



Panduan Penjual

AWS Marketplace



AWS Marketplace: Panduan Penjual

Copyright © 2025 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Merek dagang dan tampilan dagang Amazon tidak boleh digunakan sehubungan dengan produk atau layanan apa pun yang bukan milik Amazon, dengan cara apa pun yang dapat menyebabkan kebingungan di antara pelanggan, atau dengan cara apa pun yang merendahkan atau mendiskreditkan Amazon. Semua merek dagang lain yang tidak dimiliki oleh Amazon merupakan hak milik masing-masing pemiliknya, yang mungkin atau tidak terafiliasi, terkait dengan, atau disponsori oleh Amazon.

Table of Contents

Apa itu AWS Marketplace?	1
Menggunakan AWS Marketplace sebagai penjual	1
Struktur kontrak untuk AWS Marketplace	3
Harga produk di AWS Marketplace	4
Memulai	6
Persyaratan penjual untuk menerbitkan produk perangkat lunak gratis	6
Persyaratan penjual tambahan untuk produk berbayar	7
Yurisdiksi yang memenuhi syarat untuk produk berbayar	8
Portal Manajemen AWS Marketplace	15
Mendaftar sebagai AWS Marketplace penjual	15
Langkah 1: Buat profil publik Anda	17
Langkah 2: Berikan informasi pajak	17
Langkah 3: Berikan informasi rekening bank	19
Langkah 4: Tetapkan preferensi pencairan Anda	21
Langkah 5: Selesaikan proses Know Your Customer (KYC)	21
Langkah 6: Selesaikan proses verifikasi rekening bank	25
Sudah menjadi AWS Marketplace penjual?	26
Preferensi pencairan	27
Mengatur preferensi pencairan Anda	27
Penagihan untuk langganan AWS Marketplace	28
Bagaimana penjual mendapatkan pencairan mereka	28
Mata uang yang tersedia	29
Biaya listing	30
Memahami biaya untuk daftar penawaran standar di AWS Marketplace	30
Memahami biaya untuk daftar penawaran regional di AWS Marketplace	31
Keluhan Amazon Payments Eropa	31
Mengirimkan keluhan	32
Amazon Payments Europe Kerangka waktu resolusi keluhan	32
Eskalasi keluhan	32
Alat tambahan untuk AWS Marketplace penjual	33
AWS Marketplace Layanan Analisis Perdagangan	34
AWS Marketplace Program Demonstrasi Lapangan	54
AWS Asisten Mitra	55
Mempersiapkan produk Anda	60

Pengiriman produk	60
Penunjukan untuk produk yang berjalan AWS	65
Harga produk	65
Model penetapan harga	66
Mengubah model harga	71
Mengubah harga	71
Penawaran privat	72
Pengembalian uang produk	72
Wilayah dan negara	77
AWS Daerah	78
Negara-negara	78
Terjemahan dan bahasa	79
Memulai	79
Memilih keluar dari terjemahan	80
Kontrak standar	81
Kontrak Standar untuk AWS Marketplace	81
Kontrak Reseller untuk AWS Marketplace	83
Kategori dan metadata	84
Penamaan dan mendeskripsikan produk Anda	85
Memilih kategori dan kata kunci	86
Petunjuk penggunaan produk AMI dan kontainer	87
Persyaratan	88
Menulis catatan rilis	89
Menulis instruksi penggunaan	89
Menulis instruksi peningkatan	89
Menulis instruksi CloudFormation pengiriman	90
Optimisasi mesin pencari	90
Optimisasi mesin pencari	91
AWS Marketplace pencarian	92
Media promosi	94
Praktik terbaik untuk media promosi	94
Kiat untuk menambah dan mengelola media promosi	96
Mempersiapkan penawaran pribadi Anda	98
Bagaimana penawaran pribadi bekerja	99
Pertimbangan penawaran pribadi	99
Pengalaman penawaran pribadi untuk pembeli	100

Pelaporan untuk penawaran pribadi	101
Jenis produk yang didukung	102
Penawaran pribadi untuk produk AMI	102
Penawaran pribadi untuk produk kontainer	103
Penawaran pribadi untuk produk layanan profesional	104
Penawaran pribadi untuk produk SaaS	104
Penawaran pribadi untuk produk ML	104
Membuat dan mengelola penawaran pribadi	105
Memulai penawaran pribadi baru	106
Memahami status penawaran	107
Menyusun dan menerbitkan penawaran pribadi	107
Menambahkan penawaran pribadi dan tombol permintaan demo	109
Mengirim penawaran pribadi ke pembeli	112
Mengkloning penawaran pribadi Anda	113
Mengunduh detail penawaran	113
Menyimpan kemajuan penawaran pribadi Anda	114
Memperbarui berakhirnya penawaran pribadi	114
Membatalkan penawaran pribadi	115
Penawaran pribadi mitra saluran	115
Informasi tambahan	117
Membuat otorisasi penjualan sebagai ISV	117
Membuat rencana angsuran	122
Membuat paket angsuran untuk penawaran pribadi	123
Pelaporan rencana angsuran	124
Perjanjian bertanggal di masa depan	124
Pertimbangan untuk perjanjian data future	125
Membuat perjanjian bertanggal masa depan	126
Menggunakan rencana angsuran dengan perjanjian bertanggal masa depan	126
Menerima pemberitahuan untuk perjanjian bertanggal masa depan	126
Menggunakan perjanjian bertanggal future dengan penjualan kembali untuk penawaran pribadi Channel Partner	127
FAQ penawaran pribadi	128
Apa itu Tim Sukses Penawaran Pribadi (POST), dan bagaimana penjual dapat menghubungi mereka?	129
Bagaimana penjual dapat mengatasi kesalahan saat mengakses tab Penawaran?	129
Apa itu rencana angsuran atau jadwal pembayaran yang fleksibel?	130

Bagaimana saya bisa membantu pembeli yang menerima Kesalahan 404 saat mengakses penawaran pribadi?	131
Mengapa pelanggan melihat kesalahan “Anda sudah memiliki kontrak aktif” ketika mencoba menerima penawaran pribadi?	133
Bisakah penjual atau pembeli membatalkan penawaran pribadi?	135
Bagaimana cara meminta pengembalian dana atau pembatalan kontrak?	135
Kapan pembeli ditagih?	136
Langkah apa yang harus diambil penjual setelah penawaran diterima?	136
Bagaimana cara AWS membayar penjual dan mitra?	136
Bagaimana cara AWS menilai pajak?	137
Sumber daya dan dukungan	137
Menggunakan perjanjian	139
Jenis perjanjian	139
Bekerja dengan perjanjian	139
Menemukan perjanjian	140
Menyortir perjanjian	140
Melihat detail perjanjian	140
Mengunduh detail perjanjian	140
Mengubah perjanjian	141
Jenis produk yang didukung untuk amandemen penawaran publik dan pribadi	142
Membuat peningkatan, pembaruan, dan amandemen penawaran publik dan pribadi	142
Pelaporan untuk upgrade, perpanjangan, dan amandemen	144
Produk berbasis AMI	145
Metode pengantaran produk berbasis AMI	145
Sumber daya tambahan	145
Memahami produk berbasis AMI	146
Siklus hidup produk	147
Kode produk AMI	149
Perubahan permintaan	150
Formulir Beban Produk	151
Amandemen perjanjian tahunan	152
Membuat produk berbasis AMI	153
Prasyarat	153
Memahami pengalaman swalayan	154
Buat daftar	155
Sumber daya tambahan	157

Tambahkan CloudFormation template ke produk Anda	158
Menyiapkan CloudFormationtemplat	158
Diagram arsitektur	166
Mengkonversi CloudFormation template dari produk yang ada	167
Menambahkan komponen aplikasi tanpa server	172
Mengelola produk berbasis AMI	181
Membuat permintaan perubahan	183
Memperbarui visibilitas produk	187
Menambahkan dan membatasi instance	188
Mengelola versi	190
Memperbarui informasi produk	197
Mengelola ketersediaan	199
Memperbarui EULA Anda	202
Memperbarui kebijakan pengembalian dana Anda	203
Memberikan AWS Marketplace akses ke AMI Anda	204
Menghapus produk	205
Pemecahan Masalah	207
Praktik terbaik untuk membangun AMIs	208
Mengamankan hak jual kembali	209
Membangun AMI	209
Mempersiapkan dan mengamankan AMI Anda AWS Marketplace	210
Memindai AMI Anda untuk persyaratan penerbitan	211
Memverifikasi perangkat lunak Anda berjalan di perangkat AWS Marketplace AMI	212
Harga produk AMI	214
Model harga AMI	215
AWS biaya dan biaya perangkat lunak	220
Pengukuran khusus dengan AWS Marketplace Metering Service	222
Harga kontrak untuk produk AMI	239
Menggunakan AWS License Manager	242
Menerima notifikasi Amazon SNS	257
Topik Amazon SNS: aws-mp-subscription-notification	257
Berlangganan antrian Amazon SQS ke topik Amazon SNS	258
Daftar periksa produk AMI	259
Penggunaan produk	259
Persiapan AMI	260
Windows AMIs	260

Linux AMIs	261
Formulir beban produk atau tab Produk	261
Persyaratan produk berbasis AMI	261
Kebijakan penjual produk AMI	262
Kebijakan Keamanan	262
Kebijakan arsitektur	264
Petunjuk penggunaan produk AMI	265
Kebijakan versi produk AMI	265
Kebijakan informasi pelanggan	265
Kebijakan penggunaan produk	266
Produk berbasis komponen Image Builder	268
Membuat permintaan perubahan	268
Buat permintaan perubahan dengan menggunakan layanan mandiri	269
Buat permintaan perubahan	270
Dapatkan status permintaan perubahan	271
Sumber daya tambahan	271
Memperbarui visibilitas produk	272
Perbarui visibilitas produk	272
Perbarui daftar yang diizinkan (akun pratinjau)	273
Menambahkan dan membatasi instance	273
Menambahkan sebuah instance	274
Membatasi sebuah instance	275
Mengelola versi	275
Memperbarui informasi produk	276
Mengelola ketersediaan	277
Tambahkan sebuah Wilayah AWS	277
Membatasi sebuah Wilayah AWS	278
Perbarui dukungan untuk masa depan Wilayah AWS	279
Perbarui ketersediaan menurut negara	280
Memperbarui EULA Anda	281
Memperbarui kebijakan pengembalian dana Anda	282
Menghapus produk	282
Produk berbasis kontainer	285
Mendapatkan bantuan	286
Memulai dengan Produk Kontainer	286
Siklus hidup produk	286

Prasyarat	287
Ikhtisar: Buat produk kontainer	288
Langkah 1: Buat ID produk dan kode produk untuk produk kontainer Anda	288
Langkah 2: Buat daftar awal	289
Langkah 3: Tambahkan versi awal produk Anda	290
Langkah 4: (Hanya untuk produk berbayar) Integrasikan pengukuran atau harga kontrak	290
Langkah 5: Perbarui visibilitas produk	290
Langkah selanjutnya	292
Pemindaian produk kontainer untuk masalah keamanan	292
Memperbarui informasi produk	293
Menambahkan versi produk	294
Mengelola harga produk	304
Memperbarui ketersediaan menurut negara	307
Memperbarui EULA Anda	308
Menguji dan melepaskan produk Anda	308
Persyaratan produk berbasis kontainer	310
Kebijakan Keamanan	311
Persyaratan informasi pelanggan	311
Persyaratan penggunaan produk	312
Persyaratan arsitektur	313
Petunjuk penggunaan produk kontainer	314
Persyaratan untuk produk add-on Amazon EKS	314
Harga produk kontainer	332
Model harga kontainer	333
Harga kontrak untuk produk kontainer	337
Integrasi penagihan, pengukuran, dan lisensi produk kontainer	341
Pengukuran per jam dan kustom dengan AWS Marketplace Metering Service	341
Harga kontrak dengan AWS License Manager	344
Mengkonfigurasi pengukuran dengan AWS Marketplace Metering Service	346
Mengkonfigurasi pengukuran kustom menggunakan AWS Marketplace Metering Service	358
Harga kontrak dengan AWS License Manager	371
Pemberitahuan Amazon SNS untuk produk kontainer	405
Topik Amazon SNS: aws-mp-subscription-notification	405
Berlangganan antrian Amazon SQS ke topik Amazon SNS	406
Produk machine learning	408
Memahami produk pembelajaran mesin	408

SageMaker Paket model AI	408
SageMaker Algoritma AI	409
Menerapkan model inferensi	409
Siklus hidup produk	410
Harga produk pembelajaran mesin	411
Pembatasan dan kuota layanan	414
Keamanan dan kekayaan intelektual	418
Laporan pembelajaran mesin di AWS Marketplace	419
Mempersiapkan produk Anda di SageMaker AI	420
Kemasan kode Anda ke dalam gambar	421
Mengunggah gambar Anda	445
Membuat sumber daya Amazon SageMaker AI Anda	448
Cantumkan produk Anda di AWS Marketplace	455
Prasyarat	456
Langkah 1: Buat daftar baru	459
Langkah 2: Berikan informasi produk	459
Langkah 3: Tambahkan versi produk awal	460
Langkah 4: Konfigurasi model harga	461
Langkah 5: Konfigurasi kebijakan pengembalian dana	462
Langkah 6: Konfigurasi EULA	462
Langkah 7: Konfigurasi allowlist	462
Mengelola produk Anda	463
.....	464
.....	464
.....	464
.....	465
.....	467
.....	467
.....	467
Menghapus produk	468
Membuat penawaran pribadi	468
Persyaratan dan praktik terbaik	470
Praktik terbaik umum untuk produk ML	471
Persyaratan untuk informasi penggunaan	472
Persyaratan untuk input dan output	472
Persyaratan untuk notebook Jupyter	473

Ringkasan persyaratan dan rekomendasi untuk daftar produk ML	474
Pemecahan Masalah	478
400 kesalahan saat menambahkan ARN produk	478
404 kesalahan saat menambahkan ARN produk	479
SageMaker Akses AI ditolak kesalahan saat membuat produk	480
SageMaker Kegagalan AI saat membuat produk	480
Produk SaaS	481
Memulai dengan produk SaaS	481
Prasyarat	482
Langkah selanjutnya	482
Siklus hidup produk SaaS	482
Membuat produk SaaS	483
Membuat halaman produk SaaS awal	486
Mengkonfigurasi pengaturan produk SaaS	488
Mengintegrasikan produk langganan SaaS Anda	500
Mengintegrasikan produk kontrak SaaS Anda	503
Mengintegrasikan kontrak SaaS Anda dengan produk pay-as-you-go	507
Menerapkan integrasi SaaS tanpa server	511
Merencanakan produk SaaS Anda	512
Rencanakan harga Anda	512
Rencanakan integrasi penagihan Anda	513
Rencanakan integrasi Amazon SNS Anda	513
Rencanakan bagaimana pelanggan akan mengakses produk Anda	513
Pedoman produk SaaS	514
Pedoman pengaturan produk	515
Persyaratan informasi pelanggan	515
Pedoman penggunaan produk	516
Pedoman arsitektur	517
Harga produk SaaS	522
Harga untuk langganan SaaS	523
Harga untuk kontrak SaaS	525
Membuat uji coba gratis SaaS	530
Membuat penawaran uji coba gratis SaaS	530
Membatalkan penawaran uji coba gratis SaaS	531
Orientasi pelanggan	532
Mengkonfigurasi produk SaaS Anda untuk menerima pembeli baru	533

Keamanan dan pemesanan	535
Notifikasi Amazon SNS untuk produk SaaS	535
Topik Amazon SNS: aws-mp-entitlement-notification	536
Topik Amazon SNS: aws-mp-subscription-notification	537
Berlangganan antrian SQS ke topik SNS	539
Mengakses Layanan AWS Marketplace Pengukuran dan Hak APIs	539
Pengukuran untuk penggunaan	540
Memeriksa hak	547
Daftar periksa integrasi produk SaaS	548
Pelaporan	552
Contoh kode SaaS	553
ResolveCustomercontoh kode	553
GetEntitlementcontoh kode	554
BatchMeterUsagecontoh kode	556
BatchMeterUsedengan contoh kode penandaan alokasi penggunaan (Opsional)	558
Menggunakan AWS PrivateLink	561
Pengantar	561
Mengkonfigurasi produk Anda	563
Mengirimkan produk Anda ke AWS Marketplace	563
Akses pembeli ke titik akhir VPC	564
Lampiran: Daftar Periksa	565
Produk layanan profesional di AWS Marketplace	567
Mendapatkan bantuan	568
Memulai dengan produk layanan profesional	568
Prasyarat	568
Buat produk layanan profesional	569
Buat penawaran pribadi	570
Edit informasi produk	572
Mengedit harga produk	573
Edit visibilitas produk	574
Hapus produk layanan profesional	575
Memberikan detail produk	576
Deskripsi produk	576
Sumber daya tambahan	578
Informasi Dukungan	578
Dimensi harga	578

Visibilitas produk	579
Persyaratan produk	579
Pedoman pengaturan produk	579
Persyaratan informasi pelanggan	580
Pedoman penggunaan produk	580
Pedoman arsitektur	581
Harga produk layanan profesional	582
Bagaimana penawaran pribadi bekerja	582
Produk data	584
Mengirimkan produk Anda	585
Menggunakan tab Produk	587
Persyaratan logo perusahaan dan produk	588
Persyaratan untuk mengirimkan perangkat lunak paket ulang berbayar	588
Persyaratan untuk produk dengan komponen perangkat keras	590
AWS CloudFormation produk yang diluncurkan (gratis atau berbayar) atau produk AMI berbayar berbasis penggunaan	590
Memperbarui produk Anda	595
Perubahan dan pembaruan produk	596
Waktu dan harapan	597
AMIs Mengirimkan ke AWS Marketplace	597
Pemindaian swalayan AMI	597
Kloning AMI dan penetapan kode produk	598
Daftar periksa akhir	598
Pemasaran produk Anda	601
Akademi GTM 180 hari	601
Mengumumkan ketersediaan produk Anda	602
AWS Marketplace pesan	602
Ulasan tentang AWS Marketplace	603
Menautkan ke AWS Marketplace	604
Menggunakan AWS Marketplace branding	605
Memahami Beli dengan AWS sebagai penjual	605
Beli dengan AWS prasyarat	606
.....	607
Tinjau perubahan branding	607
Beli dengan AWS pelaporan	608
Menautkan langsung ke produk Anda di AWS Marketplace	608

Siaran pers	609
AWS Marketplace pedoman penggunaan merek dagang	610
Notifikasi	612
Notifikasi email	613
Jenis peristiwa	613
Deskripsi bidang	623
Kelola notifikasi	629
EventBridge Pemberitahuan Amazon	630
Acara untuk penawaran baru	631
Acara untuk set perubahan	635
Acara untuk laporan ringkasan keamanan	637
Acara untuk pencairan	638
Laporan penjual, umpan data, dan dasbor	640
Umpan data pengiriman penjual	641
Penyimpanan dan struktur umpan data	642
Mengakses umpan data	644
Menggunakan umpan data	649
Ikhtisar tabel umpan data	650
Contoh kueri umpan data	658
Umpan data	781
Laporan penjual	824
Mengakses laporan	825
Laporan AWS Marketplace penjual yang tersedia	825
Laporan bisnis harian	826
Laporan pelanggan pelanggan harian	837
Laporan pencairan	840
Laporan pendapatan bulanan yang ditagih	850
Laporan kompensasi penjualan	860
Laporan tambahan	863
Laporan detail perjanjian	864
Dasbor penjual	865
Mengakses dasbor	825
Dasbor untuk operasi keuangan	868
Dasbor untuk operasi penjualan	903
Dasbor untuk pemasaran	922
AWS Marketplace Wawasan Vendor	941

Memahami Wawasan AWS Marketplace Vendor	942
Menyiapkan sebagai penjual	943
Buat profil keamanan	943
Unggah sertifikasi	944
Unggah penilaian diri	946
Aktifkan penilaian AWS Audit Manager otomatis	947
Melihat profil Anda	953
Lihat profil keamanan Anda sebagai penjual	954
Mengelola snapshot	955
Buat snapshot	956
Lihat snapshot	956
Ekspor snapshot	957
Lihat snapshot terbaru yang dirilis	957
Tunda rilis snapshot	957
Ubah preferensi untuk daftar snapshot	958
Mengendalikan akses	958
Izin untuk AWS Marketplace penjual Wawasan Vendor	959
CreateDataSource	960
DeleteDataSource	960
GetDataSource	960
UpdateDataSource	961
ListDataSources	961
CreateSecurityProfile	961
ListSecurityProfiles	961
GetSecurityProfile	962
AssociateDataSource	962
DisassociateDataSource	962
UpdateSecurityProfile	962
ActivateSecurityProfile	962
DeactivateSecurityProfile	963
UpdateSecurityProfileSnapshotCreationConfiguration	963
UpdateSecurityProfileSnapshotReleaseConfiguration	963
ListSecurityProfileSnapshots	964
GetSecurityProfileSnapshot	964
TagResource	964
UntagResource	964

ListTagsForResource	965
Sumber daya tambahan	965
Keamanan	262
IAM untuk AWS Marketplace	967
Membuat pengguna	969
Membuat atau menggunakan grup	970
Masuk sebagai pengguna	971
Kebijakan dan izin untuk penjual AWS Marketplace	972
Kebijakan untuk AWS Marketplace penjual	973
Izin untuk penjual AWS Marketplace	974
Contoh 1: Izin untuk melihat status KYC	977
Contoh 2: Izin untuk membuat upgrade dan perpanjangan untuk penawaran pribadi	978
Contoh 3: Izin untuk mengakses halaman Penawaran dan membuat penawaran pribadi baru	979
Contoh 4: Izin untuk mengakses halaman Pengaturan	979
Contoh 5: Izin untuk mengakses halaman Unggah File	980
Menggunakan grup IAM	980
AWS kebijakan terkelola	981
AWSMarketplaceAmiIngestion	982
AWSMarketplaceFullAccess	982
AWSMarketplaceGetEntitlements	983
AWSMarketplaceMeteringFullAccess	983
AWSMarketplaceMeteringRegisterUsage	983
AWSMarketplaceSellerFullAccess	983
AWSMarketplaceSellerProductsFullAccess	987
AWSMarketplaceSellerProductsReadOnly	989
AWSMarketplaceSellerOfferManagement	991
AWSVendorInsightsVendorFullAccess	991
AWSVendorInsightsVendorReadOnly	991
Pembaruan kebijakan	991
AWS Marketplace Izin akun Layanan Analisis Perdagangan	997
Izin Amazon SQS	998
AWS Marketplace izin API pengukuran dan hak	1000
Kebijakan IAM untuk produk SaaS	1000
Kebijakan IAM untuk produk AMI	1001
Kebijakan IAM untuk produk kontainer	1001

Peran terkait layanan untuk Otorisasi Penjualan Kembali	1002
Izin peran terkait layanan untuk AWS Marketplace	1003
Membuat peran terkait layanan untuk AWS Marketplace	1006
Mengedit peran terkait layanan untuk AWS Marketplace	1006
Menghapus peran terkait layanan untuk AWS Marketplace	1006
Wilayah yang Didukung untuk AWS Marketplace peran terkait layanan	1007
Pencatatan panggilan AWS Marketplace API dengan AWS CloudTrail	1007
AWS Marketplace Mengukur contoh entri file log API	1008
Pemecahan Masalah	1014
Pesan kesalahan pendaftaran penjual umum	1014
Riwayat dokumen	1017
AWS Glosarium	1043
.....	mxliv

Apa itu AWS Marketplace?

AWS Marketplace adalah katalog digital yang dikuratori yang dapat digunakan pelanggan untuk menemukan, membeli, menyebarkan, dan mengelola perangkat lunak, data, dan layanan pihak ketiga untuk membangun solusi dan menjalankan bisnis mereka. AWS Marketplace mencakup ribuan daftar perangkat lunak dari kategori populer seperti keamanan, aplikasi bisnis, pembelajaran mesin, dan produk data di seluruh industri tertentu, seperti perawatan kesehatan, layanan keuangan, dan telekomunikasi. Pelanggan dapat dengan cepat meluncurkan perangkat lunak yang telah dikonfigurasi sebelumnya, dan memilih solusi perangkat lunak di Amazon Machine Images (AMIs), perangkat lunak sebagai layanan (SaaS), dan format lainnya. Layanan profesional juga tersedia untuk membantu pelanggan mengonfigurasi, menyebarkan, dan mengelola perangkat lunak pihak ketiga. Untuk daftar lengkap metode pengiriman, lihat [Pengiriman produk](#).

Anda dapat menggunakan AWS Marketplace sebagai pembeli (pelanggan), penjual (penyedia), atau keduanya. Siapa pun yang memiliki Akun AWS dapat menggunakan AWS Marketplace sebagai pembeli, dan dapat mendaftar untuk menjadi penjual. Penjual dapat berupa vendor perangkat lunak independen (ISV), mitra saluran, penyedia layanan terkelola (MSP), atau individu yang memiliki sesuatu untuk ditawarkan yang bekerja dengan AWS produk dan layanan.

Note

Penyedia produk data harus memenuhi persyaratan kelayakan AWS Data Exchange. Untuk informasi selengkapnya, lihat [Menyediakan produk data di AWS Data Exchange](#) di Panduan Pengguna AWS Data Exchange.

Mitra yang memenuhi syarat dapat secara terprogram membuat daftar AWS Marketplace produk di luar. AWS Marketplace Untuk informasi lebih lanjut tentang menjadi mitra yang memenuhi syarat, hubungi mitra pengembangan AWS Marketplace bisnis Anda.

Video berikut menjelaskan lebih lanjut tentang penjualan AWS Marketplace.

[Pengantar AWS Marketplace](#)

Menggunakan AWS Marketplace sebagai penjual

Proses penjualan produk perangkat lunak AWS Marketplace melibatkan tujuh langkah berikut.

Proses penjual

Langkah	Tindakan	Deskripsi
1	Daftar	Sebagai penjual, Anda mulai dengan mendaftar untuk Portal Manajemen AWS Marketplace. Kami menyarankan Anda menerapkan dedicated baru Akun AWS yang dapat dengan mudah dihubungkan dengan AWS organisasi yang sudah ada. Verifikasi bahwa informasi pajak AWS Mitra memenuhi kriteria kelayakan yurisdiksi. Untuk menerima pencairan, Anda harus memberikan rekening bank.
2	Tentukan jenis produk	<p>Tentukan jenis produk yang ingin Anda jual. Untuk informasi selengkapnya tentang membuat jenis produk di AWS Marketplace, lihat berikut ini:</p> <ul style="list-style-type: none"> • Produk berbasis AMI di AWS Marketplace • Produk berbasis kontainer pada AWS Marketplace • Produk pembelajaran mesin di AWS Marketplace • Produk berbasis SaaS di AWS Marketplace • Produk layanan profesional di AWS Marketplace • Produk data (Untuk informasi selengkapnya tentang produk data, lihat Apa itu AWS Data Exchange? dalam Panduan Pengguna AWS Data Exchange.)
3	Siapkan produk	Konfigurasi paket Anda, tetapkan skema harga, tentukan kategori yang relevan untuk mencantumkan produk Anda, dan tambahkan kata kunci sehingga produk Anda muncul dalam penelusuran yang relevan. Untuk menyederhanakan proses pengadaan, Anda dapat menggunakan persyaratan lisensi standar untuk daftar produk publik dan penawaran pribadi.
4	Kirim produk	Gunakan proses pengiriman produk untuk membuat produk Anda tersedia di AWS Marketplace. Produk bisa sederhana, misalnya, satu Amazon Machine Image (AMI) yang memiliki satu struktur harga. Atau, produk bisa rumit, dengan banyak AMIs, AWS

Langkah	Tindakan	Deskripsi
		CloudFormation templat, dan opsi harga dan jadwal pembayaran yang kompleks.
5	Produk pasar	Berkontribusi pada keberhasilan produk Anda dengan mendorong kesadaran AWS Marketplace dan dengan mengarahkan lalu lintas langsung ke halaman produk Anda AWS Marketplace.
6	Lihat laporan dan umpan data	Setelah Anda terdaftar sebagai penjual, gunakan Portal Manajemen AWS Marketplace untuk mengakses laporan penggunaan untuk produk Anda. AWS Marketplace menyediakan alat untuk mengumpulkan dan menganalisis informasi tentang penjualan produk Anda.
7	Kelola produk	Gunakan Portal Manajemen AWS Marketplace untuk mengelola akun dan halaman produk Anda.

Sebagai penjual, pergi ke [Portal Manajemen AWS Marketplace](#) untuk mendaftar. Jika Anda mengenakan biaya untuk penggunaan produk Anda, Anda juga harus memberikan informasi pajak dan perbankan sebagai bagian dari pendaftaran Anda. Ketika Anda mendaftar, Anda membuat profil untuk perusahaan Anda atau untuk diri Anda sendiri yang dapat ditemukan di AWS Marketplace. Anda juga menggunakan [Portal Manajemen AWS Marketplace](#) untuk membuat dan mengelola halaman produk untuk produk Anda.

Struktur kontrak untuk AWS Marketplace

Penggunaan perangkat lunak, layanan, dan produk data yang dijual AWS Marketplace diatur oleh perjanjian antara pembeli dan penjual. AWS bukan pihak dalam perjanjian ini.

Sebagai penjual, perjanjian Anda meliputi:

- Perjanjian lisensi pengguna akhir (EULA) Anda dengan pembeli, yang terletak di halaman daftar produk untuk daftar perangkat lunak publik di AWS Marketplace. Banyak penjual menggunakan [Standard Contract for AWS Marketplace \(SCMP\)](#) sebagai EULA default mereka. Anda juga dapat menggunakan SCMP sebagai dasar untuk negosiasi dalam penawaran pribadi dan menggunakan templat amandemen untuk memodifikasi SCMP. Penawaran pribadi juga dapat mencakup ketentuan kontrak khusus yang dinegosiasikan antara para pihak.

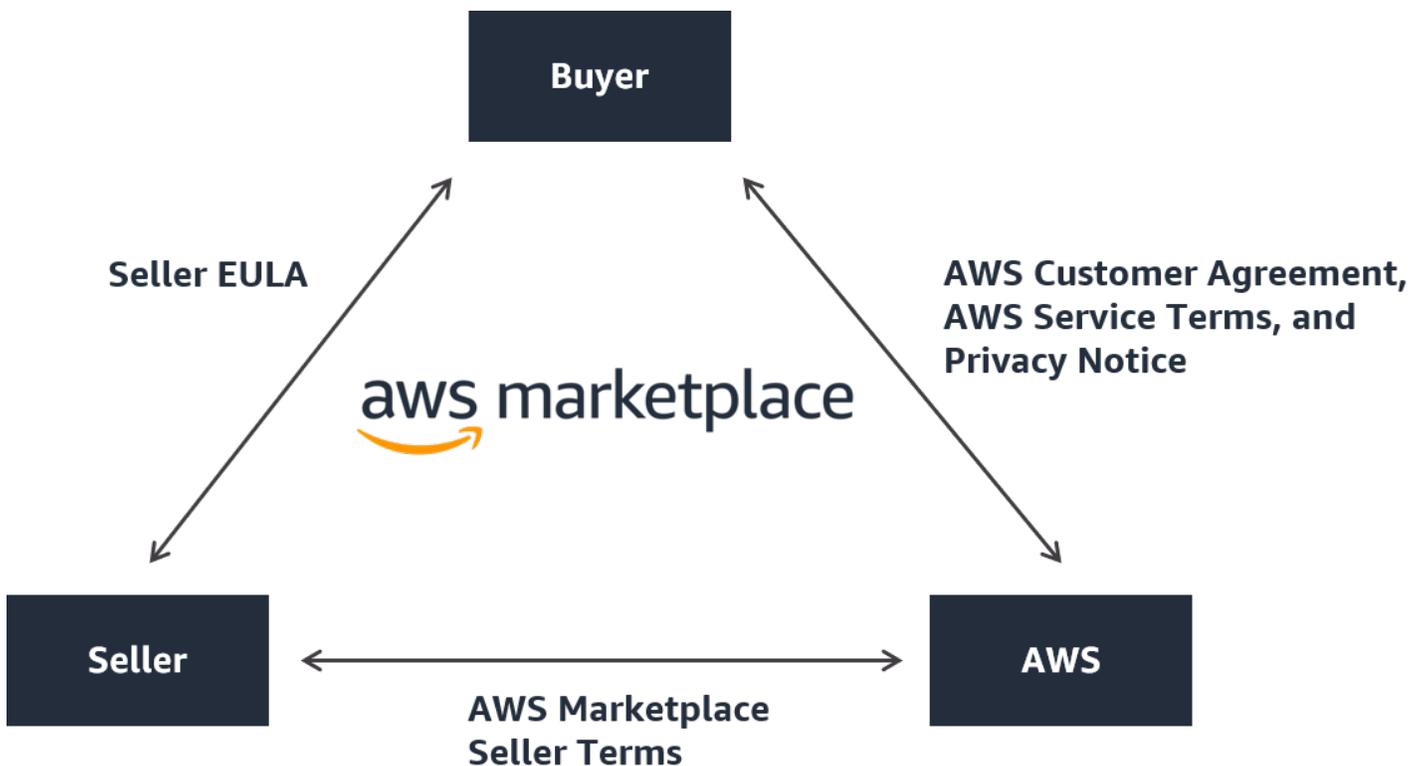
Note

Untuk informasi tentang kapan pembaruan EULA akan terjadi, berdasarkan jenis penawaran dan model harga, lihat [pembaruan EULA di Panduan AWS Marketplace Pembeli](#).

- [Ketentuan Layanan untuk AWS Marketplace Penjual](#), yang mengatur aktivitas Anda di AWS Marketplace.

Penggunaan pembeli AWS Marketplace diatur oleh [Ketentuan Layanan AWS](#), [Perjanjian AWS Pelanggan](#), dan [Pemberitahuan Privasi](#).

Grafik berikut menunjukkan struktur kontrak untuk AWS Marketplace.



Harga produk di AWS Marketplace

Di AWS Marketplace, produk dapat gratis untuk digunakan atau dapat memiliki biaya terkait. Biaya menjadi bagian dari AWS tagihan pembeli, dan setelah pembeli membayar, AWS membayar penjual. Produk bisa mengambil banyak bentuk. Misalnya, produk dapat digunakan sebagai Amazon Machine

Image (AMI) yang dipakai menggunakan pembeli. Akun AWS Produk juga dapat dikonfigurasi untuk menggunakan CloudFormation templat untuk pengiriman ke pembeli. Produk juga dapat berupa fasilitas SaaS dari ISV, daftar kontrol akses web (web ACL), seperangkat aturan, atau ketentuan untuk. AWS WAF Produk juga dapat berupa layanan profesional dari ISV, mitra saluran, atau MSP.

Opsi harga yang fleksibel termasuk uji coba gratis, per jam, bulanan, tahunan, multi-tahun, dan model Bring Your Own License (BYOL), dan ditagih dari satu sumber. AWS menangani penagihan dan pembayaran, dan biaya muncul pada tagihan pelanggan AWS .

Produk perangkat lunak dapat dibeli dengan harga yang tercantum menggunakan perjanjian lisensi pengguna akhir standar ISV (EULA). Selain itu, produk perangkat lunak dapat ditawarkan dengan harga khusus dan EULA melalui penawaran pribadi. Produk juga dapat dibeli berdasarkan kontrak dengan waktu tertentu atau batas penggunaan. Setelah berlangganan produk, pembeli dapat menggunakan AWS Service Catalog untuk menyalin produk dan mengelola bagaimana produk diakses dan digunakan dalam organisasi pembeli. Untuk informasi lebih lanjut tentang pengalaman pembeli, lihat <https://docs.aws.amazon.com/marketplace/latest/buyerguide/service-catalog.html>. Untuk informasi lebih lanjut tentang harga, lihat [the section called "Harga produk"](#).

Memulai sebagai AWS Marketplace penjual

Jika Anda adalah vendor perangkat lunak independen (ISV), mitra saluran, penyedia layanan terkelola (MSP), atau individu yang memiliki sesuatu untuk ditawarkan yang bekerja dengan AWS produk dan layanan, Anda dapat mendaftar sebagai penjual di AWS Marketplace. Mendaftar sebagai penjual adalah prasyarat untuk menerbitkan produk untuk dijual. AWS Marketplace Tergantung di mana Anda tinggal dan jenis produk apa yang Anda jual, AWS Marketplace memiliki persyaratan pendaftaran yang berbeda. Bagian berikut memberi Anda gambaran umum tentang proses pendaftaran penjual, persyaratan, dan alat terkait.

Untuk menjual perangkat lunak Anda AWS Marketplace, ikuti langkah-langkah berikut:

- Tinjau [persyaratan penjual untuk produk gratis](#) dan [persyaratan untuk produk berbayar](#).
- Selesaikan [proses pendaftaran penjual](#).
- Pelajari lebih lanjut tentang [langkah selanjutnya](#) dan [alat penjual](#).

Catatan

- Mendaftar sebagai AWS Marketplace penjual merupakan prasyarat untuk mencantumkan produk data di AWS Data Exchange dan menyediakannya. AWS Marketplace Untuk informasi selengkapnya tentang persyaratan ini, lihat [Menyediakan Produk Data di AWS Data Exchange](#) di Panduan Pengguna AWS Data Exchange.
- Untuk informasi tentang izin yang dibutuhkan AWS Marketplace penjual, lihat [Kebijakan dan izin untuk penjual AWS Marketplace](#).
- Untuk informasi selengkapnya tentang biaya listing produk, penjual terdaftar dapat melihat [Ketentuan AWS Marketplace Penjual](#) di Portal Manajemen AWS Marketplace.
- Untuk jawaban atas pertanyaan yang sering diajukan, lihat [FAQ AWS Marketplace Penjual](#).

Persyaratan penjual untuk menerbitkan produk perangkat lunak gratis

Terlepas dari apakah Anda mengenakan biaya untuk produk Anda ketika Anda menawarkannya AWS Marketplace, Anda menjual produk itu. Biaya untuk pelanggan adalah \$0,00, tetapi Anda dan

pelanggan menyetujui kontrak bersama untuk penggunaan produk. Jika Anda hanya menawarkan produk gratis, Anda tidak perlu memberikan informasi perbankan AWS Marketplace.

Untuk membuat dan menawarkan produk gratis di AWS Marketplace, Anda harus:

- Jual perangkat lunak siap produksi dengan fitur lengkap yang tersedia untuk umum.
- Memiliki proses dukungan pelanggan yang ditentukan dan organisasi pendukung.
- Menyediakan sarana untuk menjaga perangkat lunak diperbarui secara teratur dan bebas dari kerentanan.
- Ikuti praktik dan pedoman terbaik saat memasarkan produk Anda AWS Marketplace.
- Jadilah AWS pelanggan yang memiliki reputasi baik dan memenuhi persyaratan dalam syarat dan ketentuan untuk AWS Marketplace penjual.

Persyaratan penjual tambahan untuk produk berbayar

Jika Anda mengenakan biaya untuk produk Anda atau menawarkan produk Bring Your Own License model (BYOL), Anda juga harus memenuhi persyaratan berikut dan memberikan informasi tambahan ini:

- Anda harus menjadi penduduk tetap atau warga negara di [yurisdiksi yang memenuhi syarat](#), atau badan usaha yang terorganisir atau tergabung dalam salah satu area tersebut.
- Anda harus memberikan informasi pajak dan rekening bank. Untuk entitas yang berbasis di AS, formulir W-9 dan rekening perbankan dari bank yang berbasis di AS diperlukan. Anda harus mengaitkan informasi rekening bank Anda dengan [preferensi pencairan](#) untuk membuat penawaran publik atau pribadi.
- Penjual non-AS diwajibkan untuk memberikan (i) formulir W-8, pajak pertambahan nilai (PPN) atau nomor registrasi pajak barang dan jasa (GST), dan (ii) rekening bank dengan kode SWIFT di yurisdiksi yang memenuhi syarat. Jika perlu, Anda dapat mendaftar untuk rekening bank AS virtual dari [Hyperwallet](#).
- Untuk menyediakan produk data, Anda juga harus meminta on-boarding melalui panduan [Buat kasus](#) untuk Dukungan.
- Untuk menjual produk kepada pelanggan Akun AWS yang berbasis di negara dan wilayah di Eropa, Timur Tengah, dan Afrika (EMEA) (tidak termasuk Turki dan Afrika Selatan) melalui Amazon Web Services EMEA SARL, Anda harus [menyelesaikan](#) proses Kenali Pelanggan Anda. Selain itu:

- Anda menerima hingga dua pembayaran (untuk transaksi melalui AWS Inc. dan Amazon Web Services EMEA SARL).
- Anda mungkin dikenakan pajak atas biaya listing untuk transaksi tertentu, tergantung pada lokasi. Untuk informasi selengkapnya tentang pajak, lihat halaman bantuan [Pajak AWS Marketplace Penjual](#). Jika pajak pertambahan nilai (PPN) pada biaya listing Anda dinilai, AWS Marketplace akan memberikan faktur yang sesuai dengan pajak.
- Untuk informasi selengkapnya tentang Amazon Web Services EMEA SARL, lihat AWS EMEA Marketplace - Penjual di [situs web Amazon Web Services FAQs Europe](#).

Untuk menjual ke dalam AWS GovCloud (US) Region, penjual harus memiliki [AWS GovCloud \(US\) akun](#). Untuk detail tentang persyaratan ITAR, lihat [Panduan AWS GovCloud \(US\) Pengguna](#).

Untuk pertanyaan tentang persyaratan AWS Marketplace penjual atau proses pendaftaran, hubungi tim [Operasi AWS Marketplace Penjual](#).

Yurisdiksi yang memenuhi syarat untuk produk berbayar

Untuk menjual perangkat lunak berbayar di AWS Marketplace, Anda harus menjadi penduduk tetap atau warga negara di salah satu negara berikut atau SARs, atau badan usaha yang terorganisir atau tergabung di dalamnya:

- Australia¹
- Bahrain^{1 2}
- Kolombia^{1 2}
- Negara anggota Uni Eropa (UE) ¹
- Hong Kong SAR
- Israel^{1 2}
- Jepang³
- Selandia Baru¹
- Norwegia^{1 2}
- Qatar
- Korea Selatan
- Swiss^{1 2}
- Uni Emirat Arab (UEA) ^{1 2}

- Britania Raya (Inggris) ¹
- Amerika Serikat (US)

¹ Penjual produk berbayar di negara-negara ini harus memberikan informasi pendaftaran PPN di negara pendirian.

² Jika Anda sebagai penjual berada di negara yang sama dengan pembeli, Anda mungkin bertanggung jawab atas faktur pajak, pengumpulan, dan pengiriman uang. Silakan berkonsultasi dengan penasihat pajak Anda.

³ Mulai 1 April 2025, prosedur berikut berlaku untuk pengumpulan dan pengiriman Pajak Konsumsi Jepang (JCT) 10% dan penerbitan faktur berkualitas pajak (TQI) oleh Jepang untuk produk yang dijual pada: AWS AWS Marketplace

Skenario penjual dan pelanggan	Prosedur pengumpulan pajak dan faktur
<p>Penjual — Vendor perangkat lunak independen (ISV) dengan alamat di luar Jepang.</p> <p>Pelanggan — AWS akun dengan alamat di Jepang.</p>	<p>Menurut aturan Perpajakan Platform Tertentu, AWS Jepang akan melakukan hal berikut:</p> <ul style="list-style-type: none"> • Kumpulkan Pajak Konsumsi Jepang 10% (JCT). • Menerbitkan Faktur Berkualitas Pajak (TQI) kepada pelanggan dengan nomor JCT AWS Jepang T6011001106696. • Kirimkan Pajak Konsumsi Jepang (JCT) yang dikumpulkan ke Otoritas Pajak Jepang (JTA). <p>Untuk menghindari duplikasi, ISV harus berhenti mengumpulkan atau mengirimkan Pajak Konsumsi Jepang (JCT) dan menerbitkan Faktur Berkualitas Pajak (). TQIs</p>
<p>Penjual — Vendor perangkat lunak independen (ISV) dengan alamat di Jepang.</p> <p>Pelanggan — AWS akun dengan alamat di Jepang.</p>	<p>Menurut aturan Badan Khusus, AWS Jepang akan melakukan hal berikut:</p> <ul style="list-style-type: none"> • Kumpulkan Pajak Konsumsi Jepang 10% (JCT).

Skenario penjual dan pelanggan	Prosedur pengumpulan pajak dan faktur
	<ul style="list-style-type: none">• Menerbitkan Faktur Berkualitas Pajak (TQI) kepada pelanggan dengan nomor JCT AWS Jepang T6011001106696.• Mencairkan JCT yang dikumpulkan ke ISV. <p>ISV harus melakukan hal berikut:</p> <ul style="list-style-type: none">• Masukkan Nomor Perusahaan Jepang dan Nomor Pajak Konsumsi Jepang (JCT) mereka di Portal Manajemen Amazon Marketplace.• Gunakan informasi Faktur Berkualitas Pajak (TQI) yang tersedia di Portal Manajemen Amazon Marketplace untuk mengirimkan JCT yang dikumpulkan ke Otoritas Pajak Jepang (JTA).

Skenario penjual dan pelanggan	Prosedur pengumpulan pajak dan faktur
<p>Penawaran Pribadi Mitra Saluran (CPPO), ISV di luar Jepang</p> <p>1. Transaksi pertama</p> <ul style="list-style-type: none"> • Penjual — Vendor perangkat lunak independen (ISV) dengan alamat di luar Jepang. • Pelanggan -AWS Marketplace Channel Partner dengan alamat di Jepang. <p>2. Transaksi kedua</p> <ul style="list-style-type: none"> • Penjual — Mitra AWS Marketplace Saluran yang Sama. • Pelanggan —AWS akun dengan alamat di Jepang. 	<p>Untuk transaksi pertama, Mitra Saluran harus melaporkan Pajak Konsumsi Jepang (JCT) sesuai dengan mekanisme pajak biaya balik.</p> <p>Untuk transaksi kedua:</p> <ul style="list-style-type: none"> • Menurut aturan Badan Khusus, AWS Jepang akan melakukan hal berikut: <ul style="list-style-type: none"> • Kumpulkan Pajak Konsumsi Jepang 10% (JCT). • Menerbitkan Faktur Berkualitas Pajak (TQI) kepada pelanggan dengan nomor JCT AWS Jepang T6011001106696. • Mencairkan Pajak Konsumsi Jepang (JCT) yang dikumpulkan kepada Mitra Saluran. • Mitra Saluran harus melakukan hal berikut: <ul style="list-style-type: none"> • Masukkan Nomor Perusahaan Jepang dan Nomor Pajak Konsumsi Jepang (JCT) mereka di Portal Manajemen Amazon Marketplace. • Gunakan informasi Faktur Berkualitas Pajak (TQI) yang tersedia di Portal Manajemen Amazon Marketplace untuk mengirimkan JCT yang dikumpulkan ke Otoritas Pajak Jepang (JTA).

Skenario penjual dan pelanggan	Prosedur pengumpulan pajak dan faktur
<p>Channel Partner Private Offer (CPPO), semua penjual dan pelanggan di Jepang</p> <p>1. Transaksi pertama</p> <ul style="list-style-type: none"> • Penjual — Vendor perangkat lunak independen (ISV) dengan alamat di Jepang. • Pelanggan -AWS Marketplace Channel Partner dengan alamat di Jepang. <p>2. Transaksi kedua</p> <ul style="list-style-type: none"> • Penjual — Mitra AWS Marketplace Saluran yang Sama. • Pelanggan —AWS akun dengan alamat di Jepang. 	<p>Untuk transaksi pertama:</p> <ul style="list-style-type: none"> • Mitra Saluran menerima Faktur Berkualitas Pajak (TQI) dari ISV. • AWS Jepang tidak mengeluarkan faktur atau memungut Pajak Konsumsi Jepang (JCT). <p>Untuk transaksi kedua:</p> <ul style="list-style-type: none"> • Menurut aturan Badan Khusus, AWS Jepang akan melakukan hal berikut: <ul style="list-style-type: none"> • Kumpulkan Pajak Konsumsi Jepang 10% (JCT). • Menerbitkan Faktur Berkualitas Pajak (TQI) kepada pelanggan yang menggunakan nomor JCT AWS Jepang T6011001106696. • Mencairkan Pajak Konsumsi Jepang (JCT) yang dikumpulkan kepada Mitra Saluran. • Mitra Saluran harus melakukan hal berikut: <ul style="list-style-type: none"> • Masukkan Nomor Perusahaan Jepang dan Nomor Pajak Konsumsi Jepang (JCT) mereka di Portal Manajemen Amazon Marketplace. • Gunakan informasi Faktur Berkualitas Pajak (TQI) yang tersedia di Portal Manajemen Amazon Marketplace untuk mengirimkan JCT yang dikumpulkan ke Otoritas Pajak Jepang (JTA).

Skenario penjual dan pelanggan	Prosedur pengumpulan pajak dan faktur
<p>Channel Partner Private Offer (CPPO), ISV dan Channel Partner di luar Jepang</p> <p>1. Transaksi pertama</p> <ul style="list-style-type: none"> • Penjual — Vendor perangkat lunak independen (ISV) dengan alamat di luar Jepang. • Pelanggan —AWS Marketplace Channel Partner dengan alamat di luar Jepang. <p>2. Transaksi kedua</p> <ul style="list-style-type: none"> • Penjual — Mitra AWS Marketplace Saluran yang Sama. • Pelanggan —AWS akun dengan alamat di Jepang. 	<p>Untuk transaksi pertama, AWS Jepang tidak mengeluarkan faktur atau memungut Pajak Konsumsi Jepang (JCT) karena ini berada di luar lingkup JCT.</p> <p>Untuk transaksi kedua:</p> <ul style="list-style-type: none"> • Menurut aturan Bisnis Platform Tertentu, AWS Jepang akan melakukan hal berikut: <ul style="list-style-type: none"> • Kumpulkan Pajak Konsumsi Jepang 10% (JCT). • Menerbitkan Faktur Berkualitas Pajak (TQI) kepada pelanggan dengan nomor JCT AWS Jepang T6011001106696. • Kirimkan Pajak Konsumsi Jepang (JCT) yang dikumpulkan ke Otoritas Pajak Jepang (JTA). <p>Untuk menghindari duplikasi, Mitra Saluran harus berhenti mengumpulkan atau mengirimkan Pajak Konsumsi Jepang (JCT) dan menerbitkan Faktur Berkualitas Pajak (). TQIs</p>

Skenario penjual dan pelanggan	Prosedur pengumpulan pajak dan faktur
<p>Penawaran Pribadi Mitra Saluran (CPPO), Mitra Saluran di luar Jepang</p> <p>1. Transaksi pertama</p> <ul style="list-style-type: none"> • Penjual — Vendor perangkat lunak independen (ISV) dengan alamat di Jepang. • Pelanggan —AWS Marketplace Channel Partner dengan alamat di luar Jepang. <p>2. Transaksi kedua</p> <ul style="list-style-type: none"> • Penjual — Mitra AWS Marketplace Saluran yang Sama. • Pelanggan —AWS akun dengan alamat di Jepang. 	<p>Untuk transaksi pertama, AWS Jepang tidak mengeluarkan faktur atau memungut Pajak Konsumsi Jepang (JCT) karena ini berada di luar lingkup JCT.</p> <p>Untuk transaksi kedua:</p> <ul style="list-style-type: none"> • Menurut aturan Bisnis Platform Tertentu, AWS Jepang akan melakukan hal berikut: <ul style="list-style-type: none"> • Kumpulkan Pajak Konsumsi Jepang 10% (JCT). • Menerbitkan Faktur Berkualitas Pajak (TQI) kepada pelanggan dengan nomor JCT AWS Jepang T6011001106696. • Kirimkan Pajak Konsumsi Jepang (JCT) yang dikumpulkan ke Otoritas Pajak Jepang (JTA). <p>Untuk menghindari duplikasi, Mitra Saluran harus berhenti mengumpulkan atau mengirimkan Pajak Konsumsi Jepang (JCT) dan menerbitkan Faktur Berkualitas Pajak (). TQIs</p>

Untuk informasi selengkapnya tentang PPN, faktur, dan kewajiban pajak Anda sebagai penjual, lihat Bantuan [Pajak Layanan Web AWS Marketplace Penjual di Amazon](#).

Jika entitas bisnis tidak tergabung di salah satu negara atau SARs tercantum di atas, lihat [Sumber Daya untuk perusahaan di luar AWS Marketplace yurisdiksi](#).

Note

AWS Akun India Private Limited (AWS India), sebelumnya dikenal sebagai Amazon Internet Services Private Limited, akun tidak dapat digunakan untuk mendaftar. AWS Marketplace

Portal Manajemen AWS Marketplace

Anda dapat menggunakan [Portal Manajemen AWS Marketplace](#) untuk mengelola produk yang Anda jual AWS Marketplace. Semua penjual terdaftar dapat mengakses Portal Manajemen AWS Marketplace menggunakan AWS kredensialnya untuk akun yang mereka gunakan untuk membuat produk mereka. Akun yang Anda gunakan didefinisikan sebagai penjual catatan ketika pelanggan berlangganan produk Anda. Jika Anda memerlukan bantuan untuk menentukan akun spesifik yang merupakan penjual catatan untuk produk Anda, hubungi tim [Operasi AWS Marketplace Penjual](#).

Anda dapat menyelesaikan tugas-tugas berikut di portal:

- Daftar sebagai AWS Marketplace penjual.
- Gunakan halaman Produk untuk mengirimkan produk perangkat lunak baru dan memperbarui produk perangkat lunak yang ada.
- Pantau status permintaan Anda.
- Unggah file yang diperlukan untuk membuat dan mengelola produk perangkat lunak baru Anda.
- Kelola produk perangkat lunak Anda menjadi pendapatan saluran tambahan dengan memanfaatkan go-to-market aktivitas.
- Ukur hasil upaya pemasaran Anda dalam beberapa jam setelah peluncuran, termasuk penggunaan dan pendapatan yang didorong oleh kampanye Anda.
- Memungkinkan perwakilan layanan pelanggan untuk mengambil data pelanggan secara real time.
- Memulai pemindaian Amazon Machine Image (AMI) otomatis untuk mendeteksi kerentanan.

Note

Produk data dipublikasikan dan dikelola dari konsol AWS Data Exchange. Penyedia AWS Data Exchange dapat menggunakannya Portal Manajemen AWS Marketplace untuk mendaftar sebagai penjual, meminta AWS Data Exchange on-boarding, mengakses laporan penjual, dan mengirimkan permintaan pengembalian dana.

Mendaftar sebagai AWS Marketplace penjual

Jika Anda adalah vendor perangkat lunak independen (ISV), mitra saluran, penyedia layanan terkelola (MSP), atau individu yang memiliki sesuatu untuk ditawarkan yang bekerja dengan AWS

produk dan layanan, Anda dapat mendaftar sebagai penjual untuk. AWS Marketplace Mendaftar sebagai penjual adalah prasyarat untuk menerbitkan produk untuk dijual. AWS Marketplace Bagian berikut memandu Anda melalui langkah-langkah untuk berhasil mendaftar sebagai penjual AWS Marketplace.

Untuk mendaftar sebagai penjual AWS Marketplace, Anda dapat menggunakan akun yang sudah ada Akun AWS atau membuat akun baru. Semua AWS Marketplace interaksi terkait dengan akun yang Anda pilih. AWS Marketplace sangat merekomendasikan penggunaan peran AWS Identity and Access Management (IAM) untuk masuk ke akun Portal Manajemen AWS Marketplace daripada menggunakan kredensial akun root Anda. Untuk informasi lebih lanjut, lihat [AWS Marketplace keamanan](#). Anda juga dapat menggunakan IAM untuk mengkonfigurasi primer Anda Akun AWS untuk memungkinkan beberapa pengguna dengan berbagai izin untuk mengakses. Portal Manajemen AWS Marketplace Untuk informasi selengkapnya, lihat [the section called “IAM untuk AWS Marketplace”](#).

Note

Untuk menerima pemberitahuan penting dan pembaruan akun, Anda harus memberikan alamat email yang dapat diakses oleh kontak yang sesuai dalam organisasi Anda. Alias tidak dapat diganti dengan alamat email karena beberapa AWS tim hanya dapat menerima email dari alamat email yang terdaftar.

Untuk mendaftar sebagai penjual, selesaikan langkah-langkah berikut.

Topik

- [Langkah 1: Buat profil publik Anda](#)
- [Langkah 2: Berikan informasi pajak](#)
- [Langkah 3: Berikan informasi rekening bank](#)
- [Langkah 4: Tetapkan preferensi pencairan Anda](#)
- [Langkah 5: Selesaikan proses Know Your Customer \(KYC\)](#)
- [Langkah 6: Selesaikan proses verifikasi rekening bank](#)
- [Sudah menjadi AWS Marketplace penjual?](#)

Langkah 1: Buat profil publik Anda

Langkah pertama untuk mendaftar adalah memilih yang akan Akun AWS digunakan sebagai AWS Marketplace akun utama Anda, dan memberikan informasi yang ditampilkan kepada calon pembeli di AWS Marketplace konsol. Akun ini akan menjadi penjual catatan untuk produk Anda AWS Marketplace dan akan digunakan untuk pelaporan, pencairan, dan komunikasi dari AWS Marketplace Anda.

Setelah Anda menggunakan Akun AWS untuk mendaftar sebagai penjual dan membuat daftar produk AWS Marketplace, Anda tidak dapat mengubah akun yang terkait dengan produk. Kami menyarankan Anda menggunakan akun baru untuk mendaftar sebagai AWS Marketplace penjual. Namun, Anda dapat menggunakan akun yang ada jika akun tersebut dibuat setelah 27 September 2017.

Untuk membuat profil publik Anda

1. Dari [Portal Manajemen AWS Marketplace](#)(AMMP), pilih Daftar sekarang, lalu masuk ke penjual Akun AWS Anda.
2. Pilih Tambahkan profil publik untuk memberikan informasi penjual Anda.

Setelah Anda menyelesaikan profil publik, Anda dapat mempublikasikan dan menjual produk gratis. Untuk menjual produk berbayar, Anda harus memberikan informasi pajak dan perbankan Anda.

Langkah 2: Berikan informasi pajak

Anda harus memberikan informasi pajak Anda, dan informasi pajak pertambahan nilai (PPN) jika berlaku, sehingga AWS Marketplace dapat secara akurat melaporkan dan memotong pajak atas penjualan produk Anda.

Untuk memberikan informasi pajak Anda

1. Masuk ke [Portal Manajemen AWS Marketplace](#), dan pilih Pengaturan.
2. Pilih Buka dasbor pajak di bagian Informasi Pembayaran.
3. Selesaikan wawancara pajak AS.
 - Untuk menjual layanan profesional AWS Marketplace, Anda harus mengisi Kuesioner Pajak untuk DAC7

- Jika Anda melihat pesan kesalahan “Lokasi wawancara pajak tidak cocok dengan lokasi bisnis,” pastikan bahwa informasi perbankan dan pajak yang disediakan di [Billing and Cost Management](#) cocok dengan apa yang dimasukkan ke dalam Portal Manajemen AWS Marketplace. Lokasi wawancara pajak Anda harus sesuai dengan lokasi bisnis.
4. Setelah Anda melengkapi informasi pajak, kembali ke halaman Pengaturan, lalu pilih Informasi PPN lengkap, jika tersedia. Pilihan ini mengarahkan Anda ke halaman Pengaturan Pajak konsol AWS Billing

Note

Bagian informasi PPN hanya tersedia jika Anda menggunakan PPN Wilayah AWS yang mendukung PPN.

Pendaftaran pajak tambahan

Jika Anda menggunakan Nomor Pendaftaran Pajak (TRN) lokal untuk penjualan di wilayah tertentu, Anda dapat menambahkannya sebagai TRN tambahan ke akun Anda. AWS Marketplace Misalnya, TRN Swiss tambahan digunakan untuk mengelola pengumpulan pajak dan faktur untuk pembeli Swiss. Alamat bisnis yang terkait dengan TRN tambahan dapat dibagikan dengan vendor perangkat lunak independen, mitra saluran, dan pembeli.

Tanpa TRN tambahan, AWS Marketplace menentukan yurisdiksi pajak menggunakan TRN primer. Untuk informasi selengkapnya, lihat [Bagaimana cara AWS menentukan lokasi akun Anda?](#)

Untuk menambahkan pendaftaran pajak tambahan

1. Masuk ke [Portal Manajemen AWS Marketplace](#).
2. Pilih Pengaturan.
3. Pada halaman Pengaturan, pilih tab Pengaturan pajak.
4. Pilih Tambahkan informasi pajak tambahan.
5. Masukkan detail pendaftaran pajak tambahan. TRN dapat berupa nomor pajak pertambahan nilai (PPN), ID PPN, nomor registrasi PPN, atau Nomor Pendaftaran Bisnis.
6. Pilih Kirim.

Akses dokumen pajak

Anda dapat mengakses dokumen pajak Anda, seperti 1099 formulir, dari Portal Manajemen AWS Marketplace

Untuk mengakses dokumen pajak Anda

1. Masuk ke [Portal Manajemen AWS Marketplace](#), dan pilih Pengaturan.
2. Buka bagian Informasi Pembayaran.
3. Pilih formulir pajak yang relevan (1099K atau DAC7).
4. Jika formulir pajak Anda tersedia, Anda dapat mengunduhnya dari halaman dasbor Pajak.

Langkah 3: Berikan informasi rekening bank

Rekening bank di yurisdiksi yang memenuhi syarat diperlukan untuk semua penjual yang ingin menjual produk berbayar. AWS Marketplace Rekening bank Anda harus dapat menerima pencairan USD.

Note

Untuk daftar negara tempat Anda dapat menawarkan produk berbayar AWS Marketplace, lihat [Yurisdiksi yang memenuhi syarat untuk produk berbayar](#).

Untuk memberikan informasi bank

1. Masuk ke [Portal Manajemen AWS Marketplace](#), dan pilih Pengaturan.
2. Pilih Informasi perbankan lengkap di bagian Informasi Pembayaran.
3. Berikan informasi yang diperlukan tentang rekening bank Anda. Anda memiliki opsi untuk menyediakan satu atau lebih rekening bank. Rekening bank ini dapat berupa akun ACH AS, rekening bank SWIFT dari yurisdiksi yang memenuhi syarat, atau akun Hyperwallet.

Note

Jika Anda belum memberikan informasi pajak Anda (dan informasi pajak pertambahan nilai, jika berlaku), Anda tidak akan dapat memberikan informasi perbankan Anda.

Jika Anda lebih suka menerima pencairan USD menggunakan rekening bank yang berbasis di AS, Hyperwallet dapat memberi Anda akun AS.

Hyperwallet adalah penyedia layanan independen yang memungkinkan Anda mentransfer dana ke rekening bank lain dalam mata uang yang didukung. Untuk waktu yang terbatas, Anda tidak akan diminta untuk membayar biaya Hyperwallet layanan tertentu sehubungan dengan AWS Marketplace pencairan.

- Dengan menambahkan detail Hyperwallet akun Anda ke akun AWS Marketplace penjual Anda, Anda setuju dan mengakui bahwa AWS Marketplace akan membagikan nama, alamat email, dan nomor akun Anda Hyperwallet untuk mengonfirmasi status Anda sebagai AWS Marketplace penjual.
- Biaya tambahan mungkin berlaku untuk penggunaan Hyperwallet layanan Anda (termasuk biaya transfer dan biaya valuta asing yang diperlukan untuk mentransfer dana ke mata uang lokal Anda), serta nilai tukar mata uang asing. Biaya Hyperwallet layanan akan dibebaskan untuk waktu yang terbatas, dan hanya sehubungan dengan AWS Marketplace pencairan hasil dari produk Berbayar Anda ke akun Anda. Hyperwallet Untuk informasi lebih lanjut, lihat bagian Biaya di Hyperwallet situs atau hubungi Hyperwallet untuk informasi lebih lanjut dan untuk meninjau biaya yang berlaku. Untuk informasi selengkapnya tentang layanan mereka, lihat [situs Hyperwallet dukungan](#).

Untuk memulai pendaftaran dengan Hyperwallet dan mendapatkan informasi rekening bank AS Anda

1. Masuk ke [Portal Manajemen AWS Marketplace](#), lalu pilih Pengaturan, lalu pilih Informasi perbankan lengkap di bagian Informasi Pembayaran.
2. Jika Anda tidak memiliki Hyperwallet akun, dan memerlukannya untuk digunakan AWS Marketplace, pilih Tidak sebagai tanggapan terhadap Apakah Anda memiliki rekening bank AS? dan Apakah Anda terdaftar Hyperwallet? Anda akan diberikan nomor identifikasi pribadi (PIN) dan tautan untuk mendaftar Hyperwallet.
3. Setelah Anda mengaktifkan Hyperwallet akun Anda, ikuti langkah-langkah yang dijelaskan di portal Hyperwallet pendaftaran untuk menyelesaikan pendaftaran dan menerima informasi akun deposit Anda.
4. Setelah Anda memperoleh akun Hyperwallet, tambahkan informasi Hyperwallet akun Anda ke akun Anda Akun AWS dengan masuk ke akun [Portal Manajemen AWS Marketplace](#). Kemudian, pilih Pengaturan, lalu pilih Informasi perbankan lengkap di bagian Informasi Pembayaran.

Note

Untuk memperbarui informasi di Hyperwallet akun Anda, hubungi tim dukungan Hyperwallet di [HyperwalletPortal](#). Lihat tab Support untuk jam dan informasi kontak.

Langkah 4: Tetapkan preferensi pencairan Anda

Pilih opsi pencairan harian atau bulanan, pilih hari mana dalam sebulan untuk menerima pencairan, dan tetapkan mata uang ke rekening bank. Untuk informasi lebih lanjut, lihat [Preferensi pencairan untuk penjual AWS Marketplace](#).

Langkah 5: Selesaikan proses Know Your Customer (KYC)

Menyelesaikan proses Know Your Customer (KYC) diperlukan bagi penjual untuk:

- Dapatkan bayaran untuk transaksi Republik Korea
- Bertransaksi menggunakan rekening bank yang berbasis di Inggris
- Sederhanakan penjualan ke pelanggan EMEA

Langkah ini opsional tetapi direkomendasikan untuk wilayah lain. KYC melibatkan penyediaan informasi tambahan tentang perusahaan Anda, poin kontak utama, kepemilikan manfaat, dan dokumentasi pendukung.

Note

Hanya pengguna terverifikasi yang dapat memproses pengembalian uang, mengubah informasi KYC, dan mengubah informasi keuangan seperti detail rekening bank. Pengguna terverifikasi adalah pengguna terverifikasi KYC dan pengguna sekunder yang menyelesaikan verifikasi.

KYC adalah persyaratan kepatuhan yang digunakan oleh lembaga keuangan dan bisnis online untuk memverifikasi identitas pelanggan mereka. Persyaratan ini disebabkan oleh:

- Pasal 5-2 Undang-Undang tentang Pelaporan dan Penggunaan Informasi Transaksi Keuangan Tertentu di Republik Korea

- Petunjuk Layanan Pembayaran (PSD 2)
- Arahan anti pencucian uang Uni Eropa yang mengatur lembaga keuangan

AWS Marketplace transaksi melalui Amazon Web Services EMEA SARL diproses melalui Amazon Payments Europe, SCA (APE), lembaga uang elektronik berlisensi di Luksemburg yang memerlukan verifikasi identitas Anda untuk menggunakan layanan pembayaran.

 Note

Rekening bank yang berdomisili di Inggris hanya dapat menerima pencairan dalam EUR dan GBP. Untuk pencairan dalam mata uang lain, Anda akan memerlukan akun di yurisdiksi yang berbeda. Selain itu, Anda akan terbatas pada penjualan melalui Amazon Web Services EMEA SARL.

Untuk menyelesaikan proses KYC

1. Di Portal AWS Marketplace Manajemen, pilih Pengaturan.
2. Di bagian Ringkasan Akun, konfirmasi bahwa Negara yang ditampilkan sudah benar.

 Note

Pilih tautan Info untuk melihat cara mengubah negara Anda.

3. Pilih Buka informasi KYC atau pilih tab Kenali pelanggan Anda (KYC) dan kemudian pilih Mulai Kepatuhan KYC dan Anda akan diarahkan kembali ke portal pendaftaran KYC.

Untuk informasi selengkapnya tentang cara informasi digunakan dan dibagikan AWS Marketplace, lihat [Pemberitahuan Privasi Amazon Payments Europe](#).

4. Pilih Ikhtisar Pergi ke KYC.
5. Pada Ikhtisar Kenali Pelanggan Anda (KYC), baca daftar informasi dan dokumentasi yang diperlukan dan kumpulkan dokumentasi yang diperlukan (jika Anda belum melakukannya). Kemudian pilih Lanjutkan ke kepatuhan KYC.
6. Masukkan detail Dasar seperti yang diarahkan. Setelah Anda meninjau Syarat & Ketentuan Amazon Payments Europe, pilih Setuju dan lanjutkan.

Ketika Anda melanjutkan ke halaman berikutnya atau langkah berikutnya dalam proses KYC, tindakan itu menunjukkan bahwa Anda menerima Syarat & Ketentuan Amazon Payments Europe.

Jika Anda memiliki pertanyaan, lihat Pertanyaan yang Sering Diajukan (FAQ) yang terletak di sisi kanan konsol.

7. Masukkan informasi Bisnis yang diperlukan sesuai petunjuk, lalu pilih Berikutnya.

 Note

Informasi Anda disimpan setiap kali Anda memilih Berikutnya untuk pergi ke langkah berikutnya.

8. Masukkan Titik informasi kontak yang diperlukan sesuai petunjuk, lalu pilih Berikutnya.
9. Pilih apakah pemilik Beneficial sama dengan titik kontak, tambahkan pemilik manfaat (hingga empat) jika perlu, konfirmasi penambahan Anda, lalu pilih Berikutnya.
10. Pilih apakah Perwakilan hukum sama dengan titik kontak atau pemilik manfaat. Jika perwakilan hukum adalah entitas yang berbeda, berikan informasi yang diperlukan, simpan entri Anda, lalu pilih Berikutnya.
11. Untuk dokumen Tambahan, unggah izin usaha, dokumen identitas, dan surat otorisasi (jika ada).

 Note

Jika Anda memerlukan surat otorisasi, Anda dapat menggunakan sampel berikut:

Letterhead of the company

POWER TO ACT ON BEHALF OF THE COMPANY

The undersigned ****name of Company here**** (herein after, the "Company"), duly represented by

(name, date of birth, and function) ****add full name, date of birth, and function of the signatory here****,

confirms that ****add full name of the Person of Contact here**** is authorized to open an Amazon Web

Services account with Amazon Payments, accept the User Agreement and other Policies, have access

to the Amazon Web Services account, and initiate transactions in the name and on behalf of the Company.

Dated this:

Signed by:

12. Pada Tinjau dan Kirim, tinjau dan verifikasi semua informasi yang telah Anda masukkan.

Anda dapat memilih Edit untuk kembali ke bagian sebelumnya jika perlu.

13. Pilih Kirim untuk verifikasi.

Status kepatuhan KYC Anda akan ditinjau (biasanya dalam 24 jam). Anda akan diberi tahu melalui pesan email setelah peninjauan selesai. Seluruh proses KYC biasanya memakan waktu sekitar 2 minggu.

Anda dapat kembali ke tab Pengaturan untuk melihat status kepatuhan KYC Anda pada kartu Ringkasan Akun. Untuk informasi selengkapnya tentang status KYC Anda, pilih tab Kenali Pelanggan Anda (KYC) di bawah kartu Ringkasan Akun. Ini akan ditampilkan Di bawah peninjauan sampai peninjauan selesai.

Setelah KYC Anda diverifikasi, Anda harus memberikan laporan bank pada tab Informasi pembayaran sebelum Anda dapat menerima pembayaran melalui APE.

(Opsional) Tambahkan pengguna sekunder untuk prosedur Kenali Pelanggan Anda

Note

Pengguna diminta untuk mengaktifkan otentikasi multi-faktor (MFA) untuk memperbarui informasi pencairan. Untuk informasi selengkapnya tentang MFA, lihat [Multi-Factor Authentication \(MFA\)](#) untuk IAM.

Pengguna sekunder adalah individu yang dapat mengubah informasi KYC, mengontrol aliran dana atau pengembalian uang, dan mengubah informasi keuangan seperti detail rekening bank.

Hanya pengguna sekunder yang diverifikasi KYC yang dapat melakukan pembaruan yang disebutkan di atas. Pengguna sekunder ini tunduk pada kontrol penyaringan berkelanjutan yang sama dengan pemilik akun root.

Untuk menjadi KYC terverifikasi, pengguna sekunder harus menyelesaikan prosedur di [Langkah 5: Selesaikan proses Know Your Customer \(KYC\)](#).

Untuk menambahkan pengguna sekunder untuk prosedur Kenali Pelanggan Anda

1. Minta pengguna untuk masuk ke Portal Manajemen AWS Marketplace.
2. Arahkan ke tab Pengaturan.
3. Pilih tab Kenali Pelanggan Anda (KYC) dan lihat bagian untuk informasi pengguna sekunder.
4. Pilih Lengkapi informasi pengguna sekunder.

Anda diarahkan kembali ke portal pendaftaran Pengguna Sekunder.

5. Di portal pendaftaran Pengguna Sekunder, lengkapi bidang yang diperlukan, lalu pilih Berikutnya.
6. Pada halaman Review dan Submit, unggah salinan dokumen identitas (Unggah Paspor) dan bukti alamat (Unggah Dokumen).
7. Pilih Kirim untuk Verifikasi.

Status kepatuhan KYC Anda akan ditinjau (biasanya dalam 24 jam). Anda akan diberi tahu melalui pesan email setelah peninjauan selesai. Seluruh proses KYC biasanya memakan waktu sekitar 2 minggu.

Langkah 6: Selesaikan proses verifikasi rekening bank

Untuk menerima pembayaran dari Amazon Payments Europe (APE), Anda harus memberikan informasi tambahan untuk memverifikasi rekening bank pencairan Anda yang tercantum di tab Informasi Pembayaran di Portal Manajemen AWS Marketplace

Memberikan informasi bank tambahan

Untuk memberikan informasi bank tambahan

1. Masuk ke Portal Manajemen AWS Marketplace, lalu pilih Pengaturan.
2. Pilih Perbarui informasi perbankan di bagian Informasi Pembayaran.
3. Pilih akun pencairan yang sesuai.

Status Verifikasi menampilkan Tidak Terverifikasi.

4. Pilih Verifikasi.
5. Anda akan diarahkan kembali ke portal pendaftaran Verifikasi Rekening Bank tempat Anda dapat mengunggah dan mengirimkan laporan bank Anda.

Jika Anda menggunakan solusi rekening bank Hyperwallet virtual, lihat [the section called “Unduh Pernyataan Validasi Rekening Bank Anda dari Hyperwallet”](#).

6. Di portal, pilih Unggah dokumen bank lalu pilih Kirim.

Unduh Pernyataan Validasi Rekening Bank Anda dari Hyperwallet

Untuk penjual yang menggunakan [solusi rekening bank Hyperwallet virtual](#), Anda dapat mengunduh laporan Hyperwallet bank dengan menggunakan prosedur berikut. Kemudian, Anda dapat mengunggah dokumen bank sesuai petunjuk [Langkah 6: Selesaikan proses verifikasi rekening bank](#).

Untuk mengunduh laporan bank Anda dari Hyperwallet

1. Masuk ke [Hyperwallet](#) Anda.
2. Buka halaman [Informasi Rekening Deposit](#).
3. Unduh Pernyataan Validasi Rekening Bank Anda.

Sudah menjadi AWS Marketplace penjual?

Setelah Anda terdaftar sebagai penjual, tinjau topik berikut untuk informasi lebih lanjut tentang AWS Marketplace dan langkah selanjutnya:

- Pencairan — Sebagai AWS Marketplace penjual, Anda dapat mengatur preferensi pencairan untuk menerima saldo terutang Anda dalam mata uang yang Anda pilih. Untuk informasi selengkapnya, lihat [Preferensi pencairan untuk penjual AWS Marketplace](#).
- Biaya listing — Untuk informasi selengkapnya tentang biaya listing untuk AWS Marketplace penjual, lihat [Memahami biaya listing untuk AWS Marketplace penjual](#).
- Kebijakan penanganan keluhan untuk AWS Marketplace Amazon Payments Europe (APE) - Jika Anda memiliki masalah dengan layanan yang disediakan oleh Amazon Payments Europe (APE) AWS Marketplace, beri tahu kami. Umpan balik Anda membantu kami menciptakan pengalaman yang lebih baik untuk Anda dan semua pembeli dan penjual kami. Untuk informasi selengkapnya, lihat [Mengirimkan keluhan AWS Marketplace penjual untuk Amazon Payments Europe \(\) APE](#).
- Alat penjual tambahan — AWS Marketplace memberi Anda alat penjual tambahan yang dapat Anda gunakan untuk mendapatkan lebih banyak wawasan tentang basis pelanggan Anda dan membantu Anda lebih memahami penjualan Anda. Untuk informasi selengkapnya, lihat [Alat tambahan untuk AWS Marketplace penjual](#).

- Mempersiapkan produk Anda — Setelah Anda selesai mendaftarkan akun Anda sebagai penjual, Anda dapat membuat produk untuk dijual kepada pembeli melalui AWS Marketplace. Lihat informasi yang lebih lengkap di [Mempersiapkan produk Anda untuk AWS Marketplace](#).

Preferensi pencairan untuk penjual AWS Marketplace

AWS Marketplace Penjual, termasuk vendor perangkat lunak independen (ISVs) dan Mitra Saluran, dapat mengatur preferensi pencairan untuk menerima saldo terutang mereka. Sebagai penjual, Anda dapat memilih opsi pencairan harian atau bulanan, memilih hari mana dalam sebulan untuk menerima pencairan, dan menetapkan mata uang ke rekening bank. Beberapa mata uang dapat ditetapkan ke rekening bank. Bagian ini menunjukkan cara mengatur preferensi pencairan Anda. Mereka juga memberikan informasi tentang penagihan untuk AWS Marketplace langganan dan cara AWS mencairkan pembayaran.

Note

Akun ACH dan akun Hyperwallet yang berbasis di AS hanya dapat menerima pembayaran dalam USD. Untuk pencairan non-USD, Anda harus menyediakan rekening bank SWIFT.

Topik

- [Mengatur preferensi pencairan Anda](#)
- [Penagihan untuk langganan AWS Marketplace](#)
- [Bagaimana penjual mendapatkan pencairan mereka](#)
- [Mata uang yang tersedia](#)

Mengatur preferensi pencairan Anda

Untuk mengatur preferensi pencairan Anda, ikuti langkah-langkah ini. Jika Anda baru saja mendaftar, Anda mungkin perlu menunggu dua hari kerja antara membuat profil publik Anda dan menambahkan metode pencairan Anda.

1. Masuk ke Portal Manajemen AWS Marketplace, dan pilih Pengaturan.
2. Pilih tab Informasi pembayaran.
3. Di bagian Metode pencairan, pilih Tambahkan metode pencairan. Atau, Anda dapat memilih metode pencairan dan memilih Edit untuk memperbarui pengaturan Anda.

4. Untuk Mata Uang, pilih mata uang pencairan yang Anda inginkan. Kemudian, untuk rekening Bank, pilih akun tempat Anda dapat menerima pembayaran dalam mata uang yang Anda pilih.
5. Untuk mengubah dari pencairan harian ke bulanan, pilih Bulanan dan pilih nomor antara 1-28 untuk hari di bulan yang Anda inginkan agar pencairan Anda diproses.
6. Pilih Tambahkan metode pencairan.

Note

Penjual harus mengikuti izin IAM berbutir halus untuk mengakses opsi preferensi pencairan. Untuk onboard ke izin IAM berbutir halus, lihat. [the section called “Kebijakan dan izin untuk penjual AWS Marketplace”](#)

Penagihan untuk langganan AWS Marketplace

AWS bertindak sebagai mekanisme penagihan atas nama Anda. Dua opsi pembayaran paling umum yang tersedia untuk pembeli adalah kartu kredit dan faktur.

Untuk informasi lebih lanjut tentang AWS Marketplace, perhatikan hal berikut:

- Pembelian dengan pembayaran di muka ditagih segera setelah berlangganan.
- Jadwal penagihan untuk penawaran pribadi disepakati antara pembeli dan penjual.
- Ketentuan pembayaran faktur (termasuk tanggal jatuh tempo tagihan) disepakati antara pembeli dan AWS. Persyaratan tidak diungkapkan kepada vendor.
- Penawaran pribadi menggunakan penjadwal pembayaran fleksibel harus ada di faktur sebagai opsi pembayaran.
- Anda dapat memvalidasi faktur menggunakan Dasbor Pendapatan yang [Ditagih](#). Dasbor ini merangkum faktur atas nama Anda dan memberikan visibilitas tanggal pembuatan faktur dan tanggal jatuh tempo faktur. AWS

Bagaimana penjual mendapatkan pencairan mereka

- [Metode pembayaran](#) yang valid, [rekening bank terdaftar](#), dan formulir W9 yang dikirimkan diperlukan untuk pencairan.

- [Penjual produk berbayar diharuskan untuk memberikan nomor registrasi W-8, pajak pertambahan nilai \(PPN\) atau pajak barang dan jasa \(GST\), dan rekening bank di yurisdiksi yang memenuhi syarat.](#)
- AWS mencairkan pembayaran dengan cara-cara berikut:
 - Harian — Pencairan harian terjadi ketika tersedia. Penjual harus memiliki saldo positif untuk menerima pencairan.
 - Bulanan — Penjual memilih hari dalam sebulan (1-28) untuk menerima pencairan. [Dasbor Koleksi dan Pencairan menunjukkan tanggal](#) pencairan Anda.
- AWS mencairkan pembayaran dengan menggunakan transfer Automated Clearing House (ACH) atau transfer SWIFT setelah pembeli membayar faktur.
- Dana dicairkan hanya setelah dikumpulkan dari pelanggan.
- Pembayaran membutuhkan waktu sekitar 1-2 hari kerja untuk tiba di bank penjual setelah tanggal pencairan. Waktu yang tepat tergantung pada bank dan zona waktu.
- Dasbor pencairan diperbarui dalam Portal Manajemen AWS Marketplace 3-5 hari setelah pencairan.
- Rincian tentang dana yang dicairkan dan dana yang tidak terkumpul tersedia dalam laporan pencairan, termasuk piutang terbuka.

Note

Jika pembeli pindah ke yurisdiksi yang tidak mendukung mata uang yang disepakati dalam penawaran pribadi mereka, Anda harus menegosiasikan penawaran pribadi baru dalam mata uang yang didukung. Untuk informasi selengkapnya, lihat [the section called “Penawaran privat”](#).

Mata uang yang tersedia

Penjual dapat membuat penawaran pribadi dengan harga kontrak dan menerima pencairan mereka dalam mata uang berikut:

- Dolar Amerika Serikat (USD)
- Euro (EUR)
- Pound Inggris Raya (GBP)

- Dolar Australia (AUD)
- Yen Jepang (JPY)

 Note

Semua penawaran publik dan penawaran pribadi dengan harga konsumsi hanya dapat dibuat dalam USD.

Memahami biaya listing untuk AWS Marketplace penjual

AWS Marketplace menawarkan biaya listing standar dan regional untuk AWS Marketplace penjual. Biaya listing muncul sebagai item baris tunggal dalam faktur biaya daftar yang dikeluarkan oleh AWS.

 Note

Biaya listing ini berlaku mulai 5 Januari 2024 pada tengah malam UTC.

Memahami biaya untuk daftar penawaran standar di AWS Marketplace

Pelajari tentang biaya pencatatan untuk penawaran umum standar, penawaran pribadi, penawaran pribadi mitra saluran (CPPOs), dan layanan profesional. AWS Marketplace menawarkan biaya daftar berikut untuk produk.

Biaya daftar penawaran umum

Biaya daftar untuk penawaran umum perangkat lunak dan data ditentukan oleh metode penyebaran:

- Software-as-a-service (SaaS) — 3%
- Server (Amazon Machine Image (AMI), wadah, dan pembelajaran mesin) - 20%
- AWS Data Exchange - 3%

Biaya daftar penawaran pribadi

Biaya pencatatan untuk penawaran pribadi ditentukan oleh total nilai kontrak dan apakah penawaran pribadi diperbarui dari penawaran pribadi sebelumnya atau perjanjian sebelumnya di luar: AWS Marketplace

- Kurang dari \$1 juta — 3%
- Antara \$1 juta dan kurang dari \$10 juta — 2%
- Sama dengan atau lebih besar dari \$10 juta — 1.5%
- Semua pembaruan — 1.5%

Biaya daftar penawaran pribadi mitra saluran (CPPO)

Produk CPPO memiliki kenaikan 0,5% pada biaya listing, terlepas dari jenis penawaran atau metode penyebaran. Misalnya, jika produk tersebut adalah penawaran pribadi SaaS dengan total nilai kontrak kurang dari \$1 juta, biaya pencatatan akan menjadi 3,5%.

Biaya daftar layanan profesional

Semua penawaran layanan profesional memiliki biaya listing 2,5% untuk penawaran pribadi.

Memahami biaya untuk daftar penawaran regional di AWS Marketplace

Biaya listing regional tambahan akan diterapkan pada biaya listing standar untuk transaksi Anda dengan pembeli di yurisdiksi berikut.

Note

Biaya listing regional bersifat tambahan dengan biaya listing standar. Misalnya, jika produk tersebut adalah penawaran pribadi SaaS dengan total nilai kontrak kurang dari \$1 juta yang dijual kepada pembeli di Korea Selatan, biaya pencatatan akan menjadi 4% (3% biaya daftar penawaran pribadi standar ditambah biaya daftar regional 1%).

Wilayah	Biaya listing wilayah tambahan	Tanggal efektif
Korea Selatan	1%	04/01/2025

Mengirimkan keluhan AWS Marketplace penjual untuk Amazon Payments Europe () APE

Sebagai AWS Marketplace penjual, jika Anda memiliki masalah dengan layanan yang disediakan oleh Amazon Payments Europe S.C.A (APE), Anda dapat mengajukan keluhan. Jika Anda memiliki

masalah dengan layanan yang disediakan oleh Amazon Payments Europe (APE), beri tahu kami. Umpan balik Anda membantu kami menciptakan pengalaman yang lebih baik untuk Anda dan semua pembeli dan penjual kami. Bagian berikut memberi Anda rincian tentang langkah-langkah spesifik yang diperlukan untuk mengajukan keluhan terkait Amazon Payments Europe (APE) sesuai dengan kebijakan keluhan kami. Topik ini juga memberi Anda informasi tentang kebijakan pengaduan untuk Amazon Payments Europe (APE), termasuk kerangka waktu penyelesaian dan eskalasi keluhan.

Note

Hanya keluhan khusus yang AWS Marketplace akan ditangani melalui prosedur berikut. Layanan yang disediakan oleh Amazon Payments Europe S.C.A. meliputi, antara lain, pemrosesan transaksi pembayaran, verifikasi kesalahan yang mungkin muncul dalam biaya biaya, dan pencairan dana.

Mengirimkan keluhan

Jika Anda memiliki AWS Marketplace akun di Amazon Payments Europe S.C.A., keluhan Anda akan ditangani oleh Amazon Payments Europe S.C.A.

Untuk mengajukan keluhan

1. Masuk ke akun [AWS Marketplace](#) Penjual Anda.
2. Pergi ke Hubungi Kami.
3. Pilih Marketplace Komersil, Akun Penjual, Pendaftaran.
4. Berikan detail tentang keluhan Anda dan pilih Kirim.

Amazon Payments Europe Kerangka waktu resolusi keluhan

Amazon Payments Europe S.C.A. (APE) akan menanggapi dengan pembaruan atas keluhan Anda dalam waktu 15 hari kerja setelah hari menerima keluhan Anda. Dalam keadaan luar biasa di luar kendali APE, penyelesaian pengaduan dapat diperpanjang hingga 35 hari kerja, setelah hari APE pertama kali menerima pengaduan tersebut.

Eskalasi keluhan

Jika Anda tidak puas dengan tanggapan kami, Anda dapat memilih untuk meningkatkan keluhan Anda dengan menghubungi yang berikut:

- Manajemen senior Amazon Payments Eropa

Kirimkan keluhan Anda dengan mengirimkan pesan email ke manajemen senior di <ape-management@amazon.lu>. Kami akan mempertimbangkan komentar Anda dengan hati-hati dan menanggapi dalam waktu 15 hari kerja setelah hari di mana manajemen senior menerima keluhan Anda. Dalam keadaan luar biasa di luar kendali Amazon Payment Europe, penyelesaian pengaduan dapat diperpanjang hingga 35 hari kerja, setelah hari di mana manajemen senior pertama kali menerima pengaduan.

- Commission de Surveillance du Secteur Financier (CSSF)

CSSF ini adalah otoritas yang bertanggung jawab atas pengawasan kehati-hatian perusahaan di sektor keuangan di Luksemburg. Anda dapat menghubungi CSSF di 110 Route d'Arlon L-2991 Luksemburg atau gunakan halaman Kontak di: <https://www.cssf.lu/contacts/>. Untuk mendapatkan informasi lebih lanjut mengenai CSSF dan cara menghubungi mereka, lihat [keluhan Pelanggan](#) di CSSF situs web.

- Penyelesaian Sengketa Online

Jika Anda membuka akun Anda secara online di UE, Anda mungkin juga memiliki opsi untuk merujuk keluhan Anda CSSF dengan menggunakan platform Penyelesaian Sengketa Online. Opsi ini tersedia karena Amazon Payments Europe S.C.A. menyediakan layanan keuangan dan CSSF merupakan otoritas yang bertanggung jawab atas lisensinya. Untuk informasi selengkapnya, lihat platform [Penyelesaian Sengketa Online](#) di situs web Komisi Eropa.

Alat tambahan untuk AWS Marketplace penjual

AWS Marketplace memberi Anda alat penjual yang dapat Anda gunakan untuk menjual dan mengelola produk Anda. Anda dapat menggunakan alat ini untuk mendapatkan wawasan tentang basis pelanggan Anda dan membantu lebih memahami penjualan Anda. Topik ini memberikan informasi tentang alat-alat ini dengan tautan ke sumber daya tambahan.

[Portal Manajemen AWS Marketplace](#) Ini adalah alat utama Anda untuk menjual produk AWS Marketplace. Anda dapat mengelola produk Anda menjadi pendapatan saluran tambahan dengan memanfaatkan go-to-market aktivitas yang tersedia di [Portal Manajemen AWS Marketplace](#). Untuk informasi selengkapnya, lihat [Portal Manajemen AWS Marketplace](#).

AWS Marketplace menyediakan alat penjual tambahan berikut:

- Sumber Daya Lainnya Portal Manajemen AWS Marketplace — Jika Anda membuka [Portal Manajemen AWS Marketplace](#) dan masuk, Anda dapat melihat tautan ke sumber daya tambahan di halaman beranda, di bagian Sumber Daya Marketplace. Misalnya, untuk mendapatkan dukungan untuk memasarkan produk Anda dalam 90 hari menjelang peluncuran dan 90 hari setelah peluncuran, Anda dapat melihat [Akademi GTM 180 hari](#) yang ditautkan dari Portal Manajemen AWS Marketplace beranda di bawah Marketplace Resources.
- AWS Marketplace Layanan Analisis Perdagangan — Layanan Analisis AWS Marketplace Perdagangan memungkinkan Anda mengakses data produk dan pelanggan secara terprogram. AWS Marketplace Setelah mendaftar di layanan, Anda dapat mengakses laporan penggunaan, langganan, dan penagihan melalui SDK. AWS Untuk informasi selengkapnya, lihat [Mengakses data produk dan pelanggan dengan Layanan Analisis AWS Marketplace Perdagangan](#).
- AWS Marketplace Program Demonstrasi Lapangan (FDP) — FDP memungkinkan tim AWS lapangan (AWS karyawan yang disetujui secara internal) untuk menggunakan beberapa produk dan solusi tanpa biaya. AWS Marketplace Untuk informasi selengkapnya, lihat [AWS Marketplace Program Demonstrasi Lapangan](#).
- Laporan penjual, umpan data, dan dasbor - AWS Marketplace menyediakan alat untuk mengumpulkan dan menganalisis informasi tentang penjualan produk Anda. Untuk informasi selengkapnya, lihat [Laporan penjual, umpan data, dan dasbor di AWS Marketplace](#).

Untuk informasi selengkapnya, lihat topik berikut.

Topik

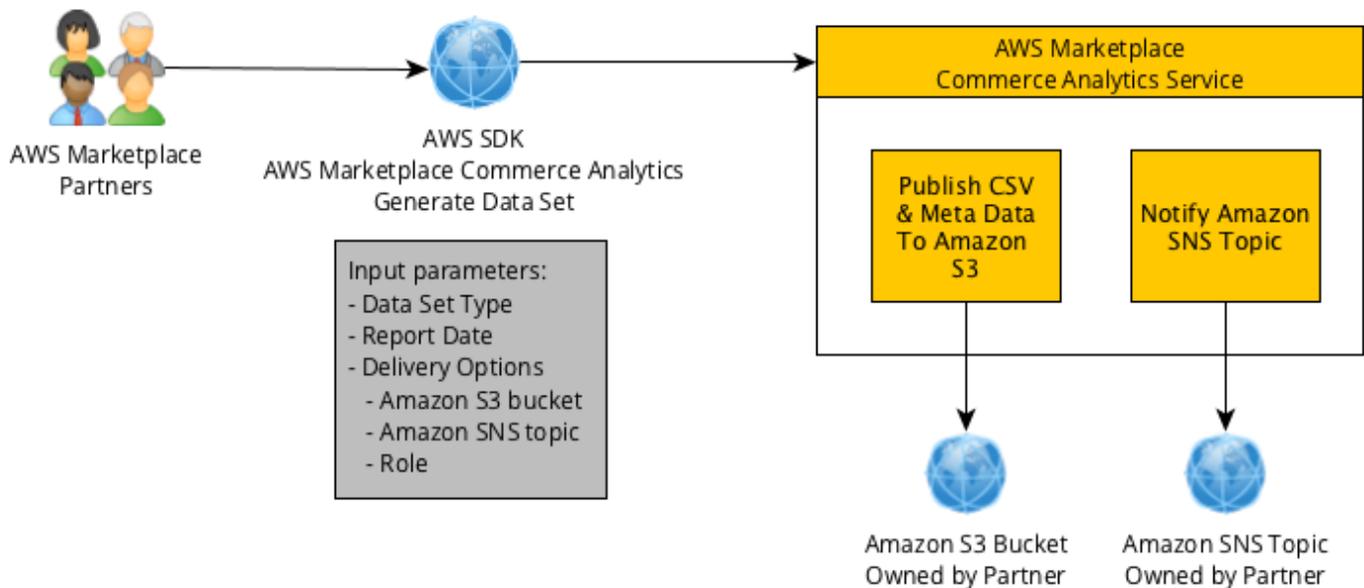
- [Mengakses data produk dan pelanggan dengan Layanan Analisis AWS Marketplace Perdagangan](#)
- [AWS Marketplace Program Demonstrasi Lapangan](#)
- [AWS Asisten Mitra](#)

Mengakses data produk dan pelanggan dengan Layanan Analisis AWS Marketplace Perdagangan

Dengan Layanan Analisis AWS Marketplace Perdagangan, Anda dapat mengakses data produk dan pelanggan secara terprogram. AWS Marketplace Setelah mendaftar di layanan, Anda dapat mengakses laporan penggunaan, langganan, dan penagihan Anda melalui AWS SDKs Data yang Anda minta menggunakan alat SDK dikirimkan ke kumpulan data Anda Akun AWS sebagai. Sebagian besar kumpulan data sesuai dengan data yang sama dengan laporan berbasis teks yang tersedia di file. [Portal Manajemen AWS Marketplace](#) Anda dapat meminta set data untuk tanggal

tertentu, dan data dikirim ke bucket Amazon S3 yang disediakan. Anda menerima pemberitahuan pengiriman data melalui Amazon Simple Notification Service (Amazon SNS). Topik ini menyediakan syarat dan ketentuan untuk menggunakan Layanan Analisis AWS Marketplace Perdagangan.

Visualisasi berikut menunjukkan cara Layanan Analisis Perdagangan mengakses data produk dan pelanggan Anda AWS Marketplace dan mengirimkannya sebagai kumpulan data ke bucket Amazon S3 Anda, memulai pemberitahuan melalui Amazon SNS.



Syarat dan Ketentuan

Syarat dan Ketentuan Layanan Analisis AWS Marketplace Perdagangan ini (“Ketentuan CAS”) berisi syarat dan ketentuan khusus untuk penggunaan dan akses Anda ke Layanan Analisis AWS Marketplace Perdagangan (“Layanan CA”) dan berlaku sejak tanggal Anda mengklik tombol “Saya Terima” atau kotak centang yang disajikan dengan Ketentuan CAS ini atau, jika sebelumnya, saat Anda menggunakan penawaran Layanan CA apa pun. Ketentuan CAS ini merupakan tambahan dari Syarat dan Ketentuan untuk AWS Marketplace Penjual (“Ketentuan AWS Marketplace Penjual”) antara Anda dan Amazon Web Services, Inc. (“AWS,” “kami,” “kami” atau “milik kami”), ketentuan-ketentuan yang dengan ini dimasukkan di sini. Jika terjadi pertentangan antara Ketentuan CAS ini dan AWS Marketplace Persyaratan Penjual, syarat dan ketentuan dari Ketentuan CAS ini berlaku, tetapi hanya sejauh konflik tersebut dan semata-mata sehubungan dengan penggunaan Layanan CA oleh Anda. Istilah huruf kapital yang digunakan di sini tetapi tidak didefinisikan di sini akan memiliki arti yang ditetapkan dalam AWS Marketplace Ketentuan Penjual.

1. Layanan CA dan Data CAS. Agar memenuhi syarat untuk mengakses Layanan CA, Anda harus menjadi AWS Marketplace Penjual yang terikat oleh Ketentuan AWS Marketplace Penjual yang ada. Informasi dan data yang Anda terima atau miliki akses sehubungan dengan Layanan CA (“Data CAS”) merupakan Informasi Pelanggan dan tunduk pada batasan dan kewajiban yang ditetapkan dalam Ketentuan AWS Marketplace Penjual. Anda dapat menggunakan Data CAS secara rahasia untuk meningkatkan dan menargetkan pemasaran dan kegiatan promosi lainnya yang terkait dengan AWS Marketplace Konten Anda asalkan Anda tidak (a) mengungkapkan Data CAS kepada pihak ketiga mana pun; (b) menggunakan Data CAS apa pun dengan cara apa pun yang tidak sesuai dengan kebijakan atau hukum privasi yang berlaku; (c) menghubungi pelanggan untuk memengaruhi mereka untuk melakukan pembelian alternatif di luar AWS Marketplace; (d) meremehkan kami, afiliasi kami, atau salah satu dari mereka atau produk kami masing-masing; atau (e) komunikasi target dalam bentuk apa pun atas dasar penerima yang dimaksud menjadi AWS Marketplace pelanggan.
2. Keterbatasan dan Keamanan Layanan CA. Anda hanya akan mengakses (atau mencoba mengakses) layanan CA dengan cara yang dijelaskan dalam dokumentasi CA Service. Anda tidak akan salah mencitrakan atau menutupi identitas Anda atau identitas klien Anda saat menggunakan Layanan CA. Kami berhak, atas kebijakan kami sendiri, untuk menetapkan dan mbucketlakukan batasan penggunaan Layanan CA oleh Anda, termasuk, tetapi tidak terbatas pada, sehubungan dengan jumlah koneksi, panggilan, dan server yang diizinkan untuk mengakses Layanan CA selama periode waktu tertentu. Anda setuju, dan tidak akan berusaha untuk menghindari keterbatasan tersebut. Kami berhak untuk membatasi, menanggguhkan, atau menghentikan hak Anda untuk mengakses Layanan CA jika kami yakin bahwa Anda mungkin melanggar Ketentuan CAS ini atau menyalahgunakan Layanan CA.
3. Kerahasiaan dan Keamanan Kredensi Layanan CA. Kredensi Layanan CA (seperti kata sandi, kunci, dan klien IDs) dimaksudkan untuk digunakan oleh Anda untuk mengidentifikasi klien API Anda. Anda sepenuhnya bertanggung jawab untuk menjaga kerahasiaan kredensialnya dan akan mengambil semua tindakan yang wajar untuk menghindari pengungkapan, deployment, atau penggunaan yang tidak sah atas mandat tersebut, termasuk, minimal, langkah-langkah yang Anda ambil untuk melindungi informasi rahasia Anda sendiri yang sifatnya serupa. CA Service kredensial mungkin tidak tertanam pada proyek open source. Anda sepenuhnya bertanggung jawab atas setiap dan semua akses ke Layanan CA dengan kredensialnya.
4. Modifikasi. Kami dapat mengubah Ketentuan CAS ini kapan saja dengan memposting versi revisi di AWS Situs atau memberi Anda pemberitahuan sesuai dengan Ketentuan AWS Marketplace Penjual. Ketentuan yang dimodifikasi akan menjadi efektif saat posting atau, jika kami memberi tahu Anda melalui email, sebagaimana tercantum dalam pesan email. Dengan terus menggunakan

atau mengakses Layanan CA setelah tanggal efektif modifikasi apa pun terhadap Ketentuan CAS ini, Anda setuju untuk terikat dengan ketentuan yang diubah.

5. Pengakhiran. Ketentuan CAS ini dan hak untuk menggunakan Data CAS yang diberikan di sini akan berakhir, dengan atau tanpa notifikasi kepada Anda setelah penghentian AWS Marketplace Penjual Syarat untuk alasan apapun. Selain itu, kami dapat berhenti menyediakan Layanan CA atau menghentikan akses Anda ke Layanan CA kapan saja tanpa alasan apa pun.

Memulai

Untuk informasi selengkapnya tentang Layanan Analisis AWS Marketplace Perdagangan, termasuk informasi orientasi, implementasi teknis, dan pemecahan masalah, lihat topik berikut.

Topik

- [Orientasi ke AWS Marketplace Layanan Analisis Perdagangan](#)
- [Menggunakan Layanan Analisis AWS Marketplace Perdagangan dengan AWS CLI dan AWS SDK untuk Java](#)
- [Menghasilkan kumpulan data dengan menggunakan Layanan Analisis AWS Marketplace Perdagangan](#)
- [Memecahkan Masalah Layanan Analisis AWS Marketplace Perdagangan](#)

Orientasi ke AWS Marketplace Layanan Analisis Perdagangan

Dengan Layanan Analisis AWS Marketplace Perdagangan, Anda dapat mengakses data produk dan pelanggan secara terprogram. AWS Marketplace Untuk memulai dengan Layanan Analisis AWS Marketplace Perdagangan, Anda harus mengonfigurasi Akun AWS dan Layanan AWS menggunakan Layanan Analisis AWS Marketplace Perdagangan. Bagian ini menunjukkan cara mengonfigurasi Akun AWS dan Layanan AWS menggunakan Layanan Analisis AWS Marketplace Perdagangan.

Untuk menggunakan Layanan Analisis AWS Marketplace Perdagangan

- [Langkah 1: Akun AWS Siapkan izin Anda](#)
- [Langkah 2: Buat bucket Amazon S3 tujuan](#)
- [Langkah 3: Konfigurasi topik Amazon SNS untuk pemberitahuan respons](#)
- [Langkah 4: Mendaftar di program Layanan Analisis Perdagangan](#)
- [Langkah 5: Verifikasi konfigurasi Anda](#)

Langkah 1: Akun AWS Siapkan izin Anda

AWS Marketplace sangat merekomendasikan penggunaan peran AWS Identity and Access Management (IAM) untuk masuk ke akun Portal Manajemen AWS Marketplace daripada menggunakan kredensial akun root Anda. Lihat [the section called “Kebijakan dan izin untuk penjual AWS Marketplace”](#) untuk izin IAM tertentu untuk AWS Marketplace Izin Layanan Commerce Analytics. Dengan membuat pengguna individu untuk orang yang mengakses akun Anda, Anda dapat memberikan setiap pengguna satu set kredensial keamanan yang unik. Anda juga dapat memberikan izin yang berbeda untuk setiap pengguna. Jika perlu, Anda dapat mengubah atau mencabut izin pengguna kapan saja.

Langkah 2: Buat bucket Amazon S3 tujuan

Layanan analitik Perdagangan mengirimkan data yang Anda minta ke bucket Amazon S3 yang Anda tentukan. Jika Anda sudah memiliki sebuah bucket Amazon S3 untuk digunakan, lanjutkan ke langkah berikutnya.

Jika Anda tidak memiliki bucket Amazon S3 atau Anda ingin membuat bucket Amazon S3 khusus untuk data ini, [lihat Bagaimana cara membuat Bucket Amazon S3](#).

Langkah 3: Konfigurasi topik Amazon SNS untuk pemberitahuan respons

Layanan Commerce Analytics memberikan notifikasi respons menggunakan Amazon SNS. Layanan menerbitkan pesan untuk topik ini untuk memberitahu Anda saat set data Anda tersedia atau jika terjadi kesalahan. Jika Anda sudah memiliki topik Amazon SNS untuk tujuan ini, lanjutkan ke langkah berikutnya.

Jika Anda tidak memiliki topik Amazon SNS yang dikonfigurasi untuk layanan ini, konfigurasi topik ini sekarang. Untuk petunjuk, lihat [Buat Topik](#).

Catat topik Amazon Resource Name (ARN) untuk topik yang Anda buat, karena ARN diperlukan untuk memanggil layanan.

Langkah 4: Mendaftar di program Layanan Analisis Perdagangan

Layanan Analytics Commerce mengakses Amazon S3 bucket dan Amazon SNS topik setelah Anda mengkonfigurasi layanan dengan ARN untuk topik dan nama bucket.

Untuk mengaktifkan akses

1. Masuk ke [Portal Manajemen AWS Marketplace](#) dengan yang Akun AWS Anda gunakan untuk mengelola AWS Marketplace produk Anda.

2. Pastikan Anda memiliki [izin IAM yang diperlukan](#) untuk mendaftar di Layanan Analisis AWS Marketplace Perdagangan.
3. Navigasikan ke [Halaman pendaftaran Layanan Analitik Perdagangan](#).
4. Masukkan nama bucket Amazon S3 dan ARN topik Amazon SNS, dan pilih Daftar.
5. Pada halaman izin, pilih Izinkan.
6. Pada Portal Manajemen AWS Marketplace, catat Nama Peran ARN dalam pesan sukses. Anda perlu ARN untuk menghubungi layanan ini.

Note

Orientasi ke Layanan Analisis Perdagangan menciptakan peran IAM dalam Anda. Akun AWS Peran IAM memungkinkan AWS Marketplace untuk menulis ke bucket Amazon S3 dan mempublikasikan pemberitahuan ke topik Amazon SNS. AWS Marketplace menggunakan akun 452565589796 untuk melakukan tindakan terkait ini dengan peran IAM ini.

Langkah 5: Verifikasi konfigurasi Anda

Langkah terakhir adalah memverifikasi bahwa konfigurasi Anda bekerja seperti yang diharapkan.

Untuk menguji konfigurasi EFA dan NCCL Anda

1. Unduh, instal, dan konfigurasi [Antarmuka Baris AWS Perintah](#) (AWS CLI).
2. Menggunakan AWS CLI, jalankan perintah ini.

```
aws marketplacecommerceanalytics generate-data-set \  
--data-set-type "customer_subscriber_hourly_monthly_subscriptions" \  
--data-set-publication-date "{TODAY'S-DATE}" \  
--role-name-arn "{YOUR-ROLE-NAME-ARN}" \  
--destination-s3-bucket-name "{amzn-s3-demo-bucket}" \  
--destination-s3-prefix "TEST_PREFIX" \  
--sns-topic-arn "{YOUR-SNS-TOPIC-ARN}"
```

- Untuk `--data-set-publication-date`, ganti `{TODAY'S DATE}` dengan tanggal saat ini menggunakan format ISO-8601, `YYYY-MM-DDT00:00:00Z`, tempat `YYYY` adalah tahun empat digit, `MM` adalah bulan dua digit, dan `DD` adalah hari dua digit.
- Untuk `--role-name-arn`, ganti `{YOUR-ROLE-NAME-ARN}` dengan ARN peran yang Anda terima dari proses pendaftaran di [Langkah 4: Mendaftar di program Layanan Analisis Perdagangan](#).
- Untuk `--destination-s3-bucket-name`, ganti `{amzn-s3-demo-bucket}` dengan nama bucket Amazon S3 yang Anda buat. [Langkah 2: Buat bucket Amazon S3 tujuan](#)
- Untuk `--sns-topic-arn`, ganti `{YOUR-SNS-TOPIC-ARN}` dengan topik Amazon SNS yang Anda buat. [Langkah 3: Konfigurasi topik Amazon SNS untuk pemberitahuan respons](#)

Jika Anda menerima tanggapan termasuk `dataSetRequestId` dari layanan, Anda telah menyelesaikan proses on-boarding. Responnya berhasil terlihat seperti ini:

```
{
  "dataSetRequestId": "646dd4ed-6806-11e5-a6d8-fd5dbcaa74ab"
}
```

Menggunakan Layanan Analisis AWS Marketplace Perdagangan dengan AWS CLI dan AWS SDK untuk Java

Dengan Layanan Analisis AWS Marketplace Perdagangan, Anda dapat mengakses data produk dan pelanggan secara terprogram. AWS Marketplace Layanan Analisis AWS Marketplace Perdagangan disediakan melalui [AWS SDK](#). Anda menggunakan [AWS CLI](#) dan [AWS SDK untuk Java](#) untuk berinteraksi dengan Layanan Analisis Perdagangan. Bagian ini menunjukkan kepada Anda cara menerapkan Layanan Analytics Commerce dengan menggunakan AWS CLI dan SDK for Java.

Topik

- [Kebijakan IAM untuk Layanan Analisis Perdagangan](#)
- [Membuat Permintaan dengan AWS CLI](#)
- [Membuat permintaan dengan AWS SDK untuk Java](#)

Kebijakan IAM untuk Layanan Analisis Perdagangan

Untuk memungkinkan pengguna Anda menggunakan Layanan Analisis Perdagangan, izin berikut diperlukan.

Gunakan kebijakan izin IAM berikut untuk mendaftar di Layanan Analisis AWS Marketplace Perdagangan.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "iam:ListRoles",
        "iam:CreateRole",
        "iam:CreatePolicy",
        "iam:AttachRolePolicy",
        "aws-marketplace-management:viewReports"
      ],
      "Resource": "*"
    }
  ]
}
```

Gunakan kebijakan izin IAM berikut untuk memungkinkan pengguna membuat permintaan ke Layanan Analisis AWS Marketplace Perdagangan.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "marketplacecommerceanalytics:GenerateDataSet",
      "Resource": "*"
    }
  ]
}
```

```
]
}
```

Untuk informasi selengkapnya, lihat [Membuat Kebijakan dalam konsol IAM](#) di Panduan Pengguna IAM.

Membuat Permintaan dengan AWS CLI

Untuk memulai, unduh aplikasi [AWS CLI](#). AWS CLI Contoh berikut membuat permintaan untuk dataset Langganan Per Jam/Bulanan untuk 1 Oktober 2017. Dataset ini diterbitkan kebucket demoBuket Amazon S3 menggunakan awalanprefiks, dan pesan notifikasi dikirim ketopik demoTopik Amazon SNS.

```
aws marketplacecommerceanalytics generate-data-set \
--data-set-type "customer_subscriber_hourly_monthly_subscriptions" \
--data-set-publication-date "2017-10-01T00:00:00Z" \
--role-name-arn "arn:aws:iam::123412341234:role/MarketplaceCommerceAnalyticsRole" \
--destination-s3-bucket-name "demo-bucket" \
--destination-s3-prefix "demo-prefix" \
--sns-topic-arn "arn:aws:sns:us-west-2:123412341234:demo-topic"
```

Permintaan ini mengembalikan sebuah identifier yang unik untuk setiap permintaan. Anda dapat menggunakan pengenal ini untuk mengkorelasikan permintaan dengan notifikasi diterbitkan untuk topik Amazon SNS Anda. Contoh berikut adalah contoh pengenal ini.

```
{
  "dataSetRequestId": "646dd4ed-6806-11e5-a6d8-fd5dbcaa74ab"
}
```

Membuat permintaan dengan AWS SDK untuk Java

Untuk memulai, unduh [AWS SDK Java](#). AWS SDK untuk Java Contoh berikut membuat permintaan untuk dataset Langganan Per Jam/Bulanan untuk 1 Oktober 2015. Dataset ini diterbitkan kebucket demoBuket Amazon S3 menggunakan awalanprefiks, dan pesan notifikasi dikirim ketopik demoTopik Amazon SNS.

```
/*
 * Copyright Amazon.com, Inc. or its affiliates. All Rights Reserved.
 *
 * Licensed under the Apache License, Version 2.0 (the "License").
 * You may not use this file except in compliance with the License.
 * A copy of the License is located at
 *
 * http://aws.amazon.com/apache2.0
 *
 * or in the "license" file accompanying this file. This file is distributed
 * on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either
 * express or implied. See the License for the specific language governing
 * permissions and limitations under the License.
 */
import java.text.DateFormat;
import java.text.ParseException;
import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.TimeZone;
import com.amazonaws.AmazonClientException;
import com.amazonaws.AmazonServiceException;
import com.amazonaws.auth.AWSCredentials;
import com.amazonaws.auth.profile.ProfileCredentialsProvider;
import com.amazonaws.regions.Region;
import com.amazonaws.regions.Regions;
import
    com.amazonaws.services.marketplacecommerceanalytics.AWSMarketplaceCommerceAnalyticsClient;
import
    com.amazonaws.services.marketplacecommerceanalytics.model.GenerateDataSetRequest;
import com.amazonaws.services.marketplacecommerceanalytics.model.GenerateDataSetResult;
/**
 * This sample demonstrates how to make basic requests to the AWS Marketplace Commerce
 * Analytics service using the AWS SDK untuk Java.
 * <p>
 * <b>Prerequisites:</b> Follow the on-boarding guide: {URL OR SOMETHING}
 * <p>
 * Fill in your AWS access credentials in the provided credentials file
 * template, and be sure to move the file to the default location
 * (~/.aws/credentials) where the sample code will load the credentials from.
 * <p>
 * <b>WARNING:</b> To avoid accidental leakage of your credentials, DO NOT keep
 * the credentials file in your source directory.

```

```
* <p>
* http://aws.amazon.com/security-credentials
*/
public class MarketplaceCommerceAnalyticsSample {
public static void main(String[] args) throws ParseException {
/*
* The ProfileCredentialsProvider will return your [default]
* credential profile by reading from the credentials file located at
* (~/.aws/credentials).
*/
AWSCredentials credentials = null;
try {
credentials = new ProfileCredentialsProvider().getCredentials();
} catch (Exception e) {
throw new AmazonClientException("Cannot load the credentials from the credential
profiles "
+ "file. Make sure that your credentials file is at the correct "
+ "location (~/.aws/credentials), and is in valid
format.", e);
}
AWSMarketplaceCommerceAnalyticsClient client = new
AWSMarketplaceCommerceAnalyticsClient(credentials);
Region usEast1 = Region.getRegion(Regions.US_EAST_1);
client.setRegion(usEast1);
System.out.println("=====");
System.out.println("Getting Started with AWS Marketplace Commerce Analytics Service");
System.out.println("=====
\n");
// Create a data set request with the desired parameters
GenerateDataSetRequest request = new GenerateDataSetRequest();
request.setDataSetType("customer_subscriber_hourly_monthly_subscriptions");
request.setDataSetPublicationDate(convertIso8601StringToDateUtc("2014-06-09T00:00:00Z"));
request.setRoleNameArn("arn:aws:iam::864545609859:role/
MarketplaceCommerceAnalyticsRole");
request.setDestinationS3BucketName("awsmp-goldmine-seller");
request.setDestinationS3Prefix("java-sdk-test");
request.setSnsTopicArn("arn:aws:sns:us-west-2:864545609859:awsmp-goldmine-seller-
topic");
System.out.println(
String.format("Creating a request for data set %s for publication date %s.",
request.getDataSetType(), request.getDataSetPublicationDate()));
try {
// Make the request to the service
GenerateDataSetResult result = client.generateDataSet(request);
```

```
// The Data Set Request ID is a unique identifier that you can use to correlate the
// request with responses on your Amazon SNS topic
System.out.println("Request successful, unique ID: " + result.getDataSetRequestId());
} catch (AmazonServiceException ase) {
System.out.println("Caught an AmazonServiceException, which means your request made it
"
+ "to the AWS Marketplace Commerce Analytics service, but was rejected with an "
+ "error response for some reason.");
System.out.println("Error Message: " + ase.getMessage());
System.out.println("HTTP Status Code: " + ase.getStatusCode());
System.out.println("AWS Error Code: " + ase.getErrorCode());
System.out.println("Error Type: " + ase.getErrorType());
System.out.println("Request ID: " + ase.getRequestId());
} catch (AmazonClientException ace) {
System.out.println("Caught an AmazonClientException, which means the client encountered
"
+ "a serious internal problem while trying to communicate with the AWS Marketplace"
+ "Commerce Analytics service, such as not being able to access the "
+ "network.");
System.out.println("Error Message: " + ace.getMessage());
}
}
private static Date convertIso8601StringToDateUtc(String dateIso8601) throws
ParseException {
    TimeZone utcTimeZone = TimeZone.getTimeZone("UTC");
    DateFormat utcDateFormat = new SimpleDateFormat("yyyy-MM-dd'T'HH:mm:ssX");
    utcDateFormat.setTimeZone(utcTimeZone);
    return utcDateFormat.parse(dateIso8601);
}
}
```

Anda harus mengharapkan hasil yang mirip dengan contoh ini.

```
=====
Getting Started with AWS Marketplace Commerce Analytics Service
=====
Creating a request for data set customer_subscriber_hourly_monthly_subscriptions for
publication
date Sun Jun 08 17:00:00 PDT 2014.
Request successful, unique ID: c59aff81-6875-11e5-a6d8-fd5dbcaa74ab
```

Menghasilkan kumpulan data dengan menggunakan Layanan Analisis AWS Marketplace Perdagangan

Setelah mendaftar di Layanan Analisis AWS Marketplace Perdagangan, Anda dapat mengakses laporan penggunaan, langganan, dan penagihan Anda melalui dan. AWS CLI AWS SDK untuk Java Data yang Anda minta menggunakan alat SDK dikirimkan ke kumpulan data Anda Akun AWS sebagai. Sebagian besar kumpulan data sesuai dengan data yang sama dengan laporan berbasis teks yang tersedia di file. [Portal Manajemen AWS Marketplace](#) Untuk informasi selengkapnya, lihat [Mengakses data produk dan pelanggan dengan Layanan Analisis AWS Marketplace Perdagangan](#). Bagian ini memberikan informasi tentang parameter, respons, dan output yang terkait dengan kumpulan data yang dihasilkan oleh Layanan Analisis Perdagangan.

Topik

- [Parameter GenerateDataSet](#)
- [Respons](#)
- [Output](#)

Parameter **GenerateDataSet**

Layanan Analisis AWS Marketplace Perdagangan memperlihatkan satu metode `GenerateDataSet`, yang dapat Anda gunakan untuk meminta kumpulan data yang akan dipublikasikan ke bucket Amazon S3 Anda. Tabel berikut mencantumkan parameter untuk `GenerateDataSet`.

Parameter basis data

Bidang	Deskripsi
Tipe Set Data	Dataset ini akan dikembalikan sebagai hasil dari permintaan.
Tanggal Publikasi Data Set	Tanggal set data diterbitkan. Untuk set data harian, berikan tanggal dengan granularitas tingkat hari untuk hari yang diinginkan.

Bidang	Deskripsi
	Untuk set data bulanan, berikan tanggal dengan granularitas tingkat bulan untuk bulan yang diinginkan. Nilai hari diabaikan.
ARN Nama Peran	ARN peran dengan kebijakan izin terlampir yang menyediakan layanan dengan akses ke sumber daya Anda.
Nama Bucket Amazon S3 Bucket	Nama (nama ramah, bukan ARN) bucket Amazon S3 tujuan. Dataset Anda dipublikasikan ke lokasi ini.
Prefiks Amazon S3 Tujuan	<p>(Opsional) prefiks Amazon S3 untuk set data diterbitkan, mirip dengan path direktori dalam sistem file standar.</p> <p>Sebagai contoh, jika diberikan nama bucketmybucketdan prefiksmyprefix/mydatasets , file output diterbitkan untuk s3://amzn-s3-demo-bucket/myprefix/mydatasets/outputfile .</p> <p>Jika struktur direktori prefiks tidak ada, ini dibuat.</p> <p>Jika tidak ada prefiks yang disediakan, set data diterbitkan ke Amazon S3 bucket root.</p>
Topik SNS ARN	ARN untuk topik Amazon SNS yang diberitahu saat set data telah diterbitkan atau jika terjadi kesalahan.

Respons

Layanan AWS Marketplace Commerce Analytics mengembalikan dua tanggapan. Yang pertama sinkron, yang segera dikembalikan, dan yang kedua adalah asynchronous, yang dikembalikan menggunakan Amazon SNS. Responnya sinkron serupa dengan contoh ini.

Adalah parameter ditetapkan?

Bidang	Deskripsi
ID Permintaan Set Data	Sebuah identifier unik yang mewakili permintaan khusus untuk layanan. Pengenal ini dapat digunakan untuk mengkorelasikan permintaan dengan notifikasi pada topik Amazon SNS.

Respon asynchronous diposting sebagai dokumen JSON diformat untuk topik Amazon SNS Anda dan mirip dengan contoh ini.

Parameter basis data

Bidang	Deskripsi
Lokasi Set Data S3	Nama bucket dan kunci untuk set data disampaikan.
Lokasi Data Set Meta Data S3	Nama bucket dan kunci untuk file metadata set data terkirim.
ID Permintaan Set Data	Sebuah identifier unik yang mewakili permintaan khusus untuk layanan. Pengenal ini dapat digunakan untuk mengkorelasikan permintaan dengan notifikasi pada topik Amazon SNS.
Sukses	“Benar” jika operasi berhasil; “palsu” jika tidak.
Pesan	(Opsional) Jika terjadi kesalahan (misalnya , “Sukses” adalah “palsu”), pesan ini berisi informasi tentang kegagalan.

Contoh respons asinkron berformat JSON

```
{
  "dataSetS3Location":{
    "bucketName":"demo-bucket",
```

```

    "key": "demo-prefix/
customer_subscriber_hourly_monthly_subscriptions_2014-06-09.csv"
  },
  "dataSetMetaDataS3Location": {
    "bucketName": "demo-bucket",
    "key": "demo-prefix/
customer_subscriber_hourly_monthly_subscriptions_2014-06-09.meta.json"
  },
  "dataSetRequestId": "f65b7244-6862-11e5-80e2-c5127e17c023",
  "success": true
}

```

Output

Setelah permintaan berhasil, set data yang diminta dikirim ke bucket Amazon S3 Anda sebagai file.csv. File metadata yang diformat JSON diterbitkan ke lokasi yang sama dengan file set data. File metadata menyediakan informasi yang berguna tentang set data dan parameter permintaan asli. File metadata memiliki nama yang sama dengan file set data, tetapi diakhiri dengan ekstensi .meta.json. Tabel berikut mencantumkan kolom metadata dalam file.csv.

Kolom metadata

Bidang	Deskripsi
ID Permintaan Set Data	Sebuah identifier unik yang mewakili permintaan khusus untuk layanan. Pengenal ini dapat digunakan untuk mengkorelasikan permintaan dengan notifikasi pada topik Amazon SNS.
Rentang Liputan Set Data	Mendefinisikan awal date/time and end date/time untuk rentang cakupan data. Tanggal ini dalam format ISO 8601.
Parameter Permintaan Set Data	Asli permintaan parameter untuk Generated ataSet metode.
Lokasi Set Data S3	Nama bucket dan kunci untuk set data disampaikan.

Bidang	Deskripsi
Lokasi Data Set Meta Data S3	Nama bucket dan kunci untuk file metadata set data terkirim.

Berikut ini adalah contoh dari isi metadata JSON-diformat.

```
{
  "dataSetRequestId": "43d7137b-8a94-4042-a09d-c41e87f371c1",
  "dataSetCoverageRange": {
    "startDateTime": "2014-06-08T00:00:00.000Z",
    "endDateTime": "2014-06-08T23:59:59.000Z"
  },
  "dataSetRequestParameters": {
    "sellerAccountId": "123412341234",
    "dataSetType": "customer_subscriber_hourly_monthly_subscriptions",
    "dataSetPublicationDate": "2014-06-09T00:00:00.000Z",
    "roleNameArn": "arn:aws:iam::123412341234:role/MarketplaceCommerceAnalyticsRole",
    "destinationS3BucketName": "demo-bucket",
    "destinationS3Prefix": "demo_prefix/customer_subscriber_hourly_monthly_subscriptions",
    "snsTopicArn": "arn:aws:sns:us-west-2:123412341234:demo-topic"
  },
  "dataSetS3Location": {
    "bucketName": "demo-bucket",
    "key": "demo_prefix/customer_subscriber_hourly_monthly_subscriptions_2014-06-09.csv"
  },
  "dataSetMetaDataSetS3Location": {
    "bucketName": "demo-bucket",
    "key": "demo_prefix/customer_subscriber_hourly_monthly_subscriptions_2014-06-09.meta.json"
  }
}
```

Untuk daftar lengkap kumpulan data yang tersedia, termasuk tanggal ketersediaan, lihat dokumentasi [AWS SDK](#).

Memecahkan Masalah Layanan Analisis AWS Marketplace Perdagangan

Anda dapat memecahkan masalah dengan Layanan Analisis AWS Marketplace Perdagangan, yang secara terprogram menyediakan data produk dan pelanggan. AWS Marketplace Anda mungkin perlu memecahkan masalah Layanan Analisis Perdagangan ketika Anda mengalami kesalahan atau masalah konfigurasi lainnya. Bagian berikut memandu Anda melalui proses pemecahan masalah, yang mencakup langkah-langkah untuk mendiagnosis dan menyelesaikan masalah umum dengan Layanan Analisis Perdagangan.

Saya tidak dapat mengakses layanan karena masalah daftar izin.

Jika Anda belum terdaftar sebagai penjual AWS Marketplace, kunjungi [Portal Manajemen AWS Marketplace](#) untuk mendaftar. Jika Anda telah terdaftar sebagai penjual AWS Marketplace, hubungi tim [Operasi AWS Marketplace Penjual](#).

Saya tidak dapat meminta kumpulan data untuk tanggal di masa lalu, meskipun dokumentasi SDK mengatakan itu harus tersedia untuk tanggal ini.

Meskipun set data terdaftar tersedia untuk tanggal tertentu di masa kemudian, kami hanya memiliki data sejak Anda bergabung AWS Marketplace. Jika Anda yakin bahwa ini salah, hubungi [AWS Marketplace Operasi Penjual](#) Tim.

Ketika saya menelepon layanan, saya menerima pesan kesalahan "Tidak dapat terhubung ke URL titik akhir: https://marketplacecom merceanalytics.eu-central-1.amazonaws.com/"

Layanan Analisis AWS Marketplace Perdagangan hanya tersedia di Wilayah AS Timur (Virginia N.). Anda harus melakukan semua panggilan ke Layanan Analitik Perdagangan ke us-east-1 Endpoint.

Jika Anda menggunakan AWS CLI, tambahkan `--region` tanda "untuk setiap panggilan dan tentukan Wilayah AWS `us-east-1`, seperti yang ditunjukkan pada contoh berikut.

```
aws marketplacecommerceanalytics generate-data-set \  
--data-set-type "customer_subscriber_hourly_monthly_subscriptions" \  
--data-set-publication-date "2016-04-21T00:00:00Z" \  
--role-name-arn "arn:aws:iam::138136086619:role/MarketplaceCommerceAnalyticsRole" \  
--destination-s3-bucket-name "marketplace-analytics-service" \  
--destination-s3-prefix "test-prefix" \  
--sns-topic-arn "arn:aws:sns:eu-  
central-1:138136086619:Marketplace_Analytics_Service_Notice" \  
--region us-east-1
```

Saya ingin menggunakan bucket Amazon S3 atau topik Amazon SNS yang berbeda dari yang saya pilih ketika saya melalui proses on-boarding.

Saat mendaftar di Layanan Analisis AWS Marketplace Perdagangan, Anda menentukan bucket Amazon S3 dan topik Amazon SNS. Proses orientasiing mengkonfigurasi izin IAM Anda untuk mengizinkan akses layanan hanya ke sumber daya tertentu ini. Untuk menggunakan sumber daya yang berbeda, Anda perlu mengubah kebijakan IAM Anda:

1. Masuk ke AWS Management Console dan buka konsol IAM di <https://console.aws.amazon.com/iam/>.
2. Pilih Peran Di sisi kiri konsol IAM.
3. Pilih MarketplaceCommerceAnalyticsRole.
4. Perluas Peran Inline bagian, jika belum diperluas.
5. Temukan kebijakan dengan nama yang dimulai oneClick_MarketplaceCommerceAnalyticsRole dan pilih Edit Kebijakan.
6. Dalam dokumen kebijakan, temukan bagian yang menentukan tindakan yang terkait dengan layanan yang ingin Anda ubah. Misalnya, untuk mengubah keranjang Amazon S3 Anda, temukan bagian yang meliputi tindakan yang dimulai dengan s3: dan mengubah masing-masing Sumber Daya Pilihan untuk menentukan bucket Amazon S3 baru.

[Untuk informasi tambahan tentang kebijakan IAM, lihat panduan berikut: _policies.html https://docs.aws.amazon.com/IAM/latest/UserGuide/access](https://docs.aws.amazon.com/IAM/latest/UserGuide/access_policies.html)

Saya mendapatkan **AccessDeniedException** kesalahan ketika saya memanggil **GenerateDataSet** tindakan

Hal ini dapat terjadi jika pengguna Anda tidak memiliki izin yang diperlukan untuk menelepon GenerateDataSet. Prosedur berikut menguraikan langkah-langkah yang diperlukan untuk membuat kebijakan IAM dengan izin tersebut menggunakan konsol IAM dan menambahkan izin ke pengguna, grup, atau peran Anda.

Cara menggunakan editor kebijakan JSON untuk membuat kebijakan

1. Masuk ke AWS Management Console dan buka konsol IAM di <https://console.aws.amazon.com/iam/>.

2. Pada panel navigasi di sebelah kiri, pilih Kebijakan.

Jika ini pertama kalinya Anda memilih Kebijakan, akan muncul halaman Selamat Datang di Kebijakan Terkelola. Pilih Memulai.

3. Di bagian atas halaman, pilih Buat kebijakan.
4. Di bagian Editor kebijakan, pilih opsi JSON.
5. Masukkan dokumen kebijakan JSON berikut:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "marketplacecommerceanalytics:GenerateDataSet",
      "Resource": "*"
    }
  ]
}
```

6. Pilih Berikutnya.

Note

Anda dapat beralih antara opsi editor Visual dan JSON kapan saja. Namun, jika Anda melakukan perubahan atau memilih Berikutnya di editor Visual, IAM dapat merestrukturisasi kebijakan Anda untuk mengoptimalkannya bagi editor visual. Untuk informasi selengkapnya, lihat [Restrukturisasi kebijakan](#) dalam Panduan Pengguna IAM.

7. Pada halaman Tinjau dan buat, masukkan Nama kebijakan dan Deskripsi (opsional) untuk kebijakan yang Anda buat. Tinjau Izin yang ditentukan dalam kebijakan ini untuk melihat izin yang diberikan oleh kebijakan Anda.
8. Pilih Buat kebijakan untuk menyimpan kebijakan baru Anda.

Untuk memberikan akses dan menambahkan izin bagi pengguna, grup, atau peran Anda:

- Pengguna dan grup di AWS IAM Identity Center:

Buat rangkaian izin. Ikuti instruksi di [Buat rangkaian izin](#) di Panduan Pengguna AWS IAM Identity Center .

- Pengguna yang dikelola di IAM melalui penyedia identitas:

Buat peran untuk federasi identitas. Ikuti instruksi dalam [Buat peran untuk penyedia identitas pihak ketiga \(federasi\)](#) dalam Panduan Pengguna IAM.

- Pengguna IAM:
 - Buat peran yang dapat diambil pengguna Anda. Ikuti instruksi dalam [Buat peran untuk pengguna IAM](#) dalam Panduan Pengguna IAM.
 - (Tidak disarankan) Lampirkan kebijakan langsung ke pengguna atau tambahkan pengguna ke grup pengguna. Ikuti petunjuk dalam [Menambahkan izin ke pengguna \(konsol\)](#) dalam Panduan Pengguna IAM.

Masalah saya tidak tercantum di sini.

Hubungi Tim [AWS Marketplace Operasi Penjual](#).

AWS Marketplace Program Demonstrasi Lapangan

Program Demonstrasi AWS Marketplace Lapangan (FDP) memungkinkan tim AWS lapangan (AWS karyawan yang disetujui secara internal) untuk menggunakan beberapa produk dan solusi tanpa biaya. AWS Marketplace

Contoh AWS karyawan yang disetujui dapat mencakup arsitek solusi dan profesional penjualan dan pemasaran. FDP memungkinkan karyawan ini untuk menunjukkan kemampuan produk untuk pendidikan dan potensi inklusi dalam beban kerja pelanggan.

Tipe produk berikut didukung:

- [Gambar Mesin Amazon \(AMIs\)](#)
- [Kontainer](#)
- [Algoritma pembelajaran mesin dan paket model \(SageMaker AI\)](#)
- [Kumpulan data \(AWS Data Exchange\)](#)

Note

Untuk produk AWS Data Exchange, FDP hanya berlaku untuk produk dengan penawaran publik sebesar \$0 (gratis).

Untuk produk AWS Data Exchange yang mengaktifkan verifikasi langganan, penyedia harus menyetujui permintaan berlangganan. Untuk informasi selengkapnya tentang

verifikasi langganan, lihat [Verifikasi langganan untuk pelanggan](#) di Panduan Pengguna AWS Data Exchange.

Anda secara otomatis terdaftar dalam program FDP ketika Anda mendaftar sebagai penjual. AWS Marketplace Untuk memilih keluar, kirimkan permintaan dukungan ke tim [Operasi AWS Marketplace Penjual](#).

Untuk melihat informasi tentang penggunaan produk di bawah program ini, lihat [AWS penggunaan demonstrasi kolombagian](#) dari [Laporan pendapatan ditagih bulanan](#).

AWS Asisten Mitra

AWS Asisten Mitra adalah chatbot bertenaga AI generatif untuk s. AWS Partner Ini dapat diakses dari Partner Central dan Portal Manajemen AWS Marketplace (AMMP) dan menggunakan Amazon Q Business, yang dilatih sesuai dengan dokumentasi yang eksklusif untuk mitra terdaftar.

Note

Untuk mengakses AWS Asisten Mitra, mitra harus menautkan akun Pusat AWS Mitra mereka dengan AWS akun mereka. Untuk informasi selengkapnya, lihat [Menautkan akun Pusat AWS Mitra dengan akun AWS Marketplace penjual](#) di Panduan Memulai Pusat AWS Mitra.

Chatbot memberikan jawaban real-time dan ringkasan dokumentasi tentang berbagai topik yang terkait dengan AWS layanan, panduan AWS Marketplace penjual, dan insentif, manfaat, dan program Jaringan AWS Mitra. Jika chatbot tidak dapat memberikan tanggapan, ia menyarankan Anda tentang cara mengajukan tiket dengan tim dukungan. Anda juga dapat memberikan umpan balik dari dalam chatbot menggunakan tombol jempol ke atas dan jempol ke bawah, yang membantu meningkatkan akurasi dan kualitas chatbot.

Berikut ini adalah interaksi khas dengan Asisten AWS Mitra:

1. Ajukan pertanyaan: Buka chatbot dari AWS Partner Central atau AMMP, dan ajukan pertanyaan menggunakan bahasa sehari-hari.
2. AWS Asisten Mitra memproses permintaan: Chatbot menggunakan pemrosesan bahasa alami (NLP) untuk menafsirkan kueri Anda.

3. AWS Asisten Mitra menghasilkan respons: Chatbot mengembalikan respons yang relevan, yang mencakup sumber daya tambahan.
4. Berikan umpan balik: Chatbot meminta Anda untuk umpan balik.

Manfaat dan fitur

AWS Asisten Mitra menyediakan opsi layanan mandiri yang meningkatkan akses ke informasi.

- Jawaban langsung: Mitra biasanya harus menghubungi AWS langsung untuk mendapatkan bantuan, tetapi Asisten AWS Mitra selalu tersedia.
- Jawaban yang akurat: Mitra menerima informasi faktual dan tepat waktu dengan variabilitas respons minimal.
- Dukungan swalayan: Mengurangi waktu yang diperlukan untuk menemukan sumber daya yang relevan.
- Tanggapan lanjutan: Memberikan respons alami, intuitif, seperti manusia.
- Pemahaman kontekstual: Memberikan informasi tambahan tentang mata pelajaran yang kompleks.
- Sistem belajar mandiri: Terus meningkat dengan belajar dari interaksi, menyempurnakan respons dari waktu ke waktu, dan memperluas basis pengetahuannya.
- Dukungan yang dapat diskalakan: Mampu menangani sejumlah besar pertanyaan secara bersamaan.

Cara menggunakan Asisten AWS Mitra

1. Masuk ke [AWS Partner Central](#) atau [AMMP](#).
2. Akses obrolan:
 - a. Pilih ikon mengambang ungu di sudut kanan bawah.
 - b. Pilih dari salah satu petunjuk yang disarankan, atau mulailah mengetik pertanyaan Anda di bidang teks.
 - c. Chatbot merespons dengan sumber dan tautan.
3. Akses chatbot dari Partner Central:
 - a. Pilih menu Support.
 - b. Pilih Obrolan instan dengan Asisten AWS Mitra dari menu tarik-turun.
 - c. (Opsional) Akses chatbot dari tiket dukungan. Sebelum mengirimkan tiket dukungan, pilih Tanya Asisten AWS Mitra untuk mendapatkan jawaban langsung.

4. Akses chatbot dari AMMP:

- a. Dari halaman Hubungi Kami, pilih Ask AWS Partner Assistant.
- b. (Opsional) Akses chatbot dari tiket dukungan. Sebelum mengirimkan tiket dukungan, pilih Kirim permintaan dukungan dari halaman Hubungi Kami. Pilih Ask AWS Partner Assistant untuk mendapatkan jawaban langsung.

5. (Opsional) Berikan umpan balik:

- a. Setelah setiap respons, Anda memiliki opsi untuk memberikan umpan balik: Pilih Thumbs Up jika respons memenuhi kebutuhan Anda.
- b. Pilih Thumbs Down jika responsnya tidak membantu.
- c. Jika Anda memilih Thumbs Down, berikan detail lebih lanjut di kotak teks atau pilih opsi dari daftar pilihan yang paling menjelaskan mengapa respons tidak membantu.
- d. Nilai pengalaman Anda dengan memilih Umpan Balik dari sudut kiri bawah jendela chatbot: Pilih Umpan Balik untuk membuka pop-up dengan skala penilaian. Gunakan bidang teks untuk komentar atau umpan balik tambahan.

Pertanyaan yang Sering Diajukan Asisten Mitra

Berikut ini adalah pertanyaan yang sering diajukan untuk Asisten AWS Mitra.

Apa itu Asisten AWS Mitra?

AWS Asisten Mitra adalah chatbot bertenaga AI generatif yang membantu mitra menemukan informasi dan menjawab pertanyaan.

Bagaimana cara mengakses Asisten AWS Mitra?

AWS Asisten Mitra tersedia untuk penjual AWS Mitra dan Marketplace, tetapi bukan pembeli Marketplace. Anda dapat mengakses chatbot dari halaman mana pun melalui AWS Partner Central atau AWS Marketplace Management Portal (AMMP).

Apakah Asisten AWS Mitra selalu tersedia?

Ya.

Bahasa apa yang didukung oleh Asisten AWS Mitra?

Saat ini, Asisten AWS Mitra hanya mendukung bahasa Inggris, tetapi akan mendukung bahasa lain di rilis mendatang.

Apa yang bisa dilakukan Asisten AWS Mitra?

Chatbot menyediakan jawaban real-time dan ringkasan dokumentasi tentang AWS layanan, panduan AWS Marketplace penjual, dan AWS Partner program, insentif, dan manfaat. Chatbot menggunakan pustaka konten yang diterbitkan yang eksklusif untuk mitra.

Apa yang bisa saya tanyakan kepada Asisten AWS Mitra?

Chatbot menyediakan jawaban real-time dan ringkasan dokumentasi tentang layanan, panduan penjual AWS Marketplace AWS, dan program Jaringan AWS Mitra, insentif, dan manfaat. Chatbot menggunakan pustaka konten yang diterbitkan yang eksklusif untuk mitra.

Masalah apa yang tidak boleh saya tanyakan kepada Asisten AWS Mitra?

Untuk masalah yang memerlukan pemecahan masalah mendalam, verifikasi akun, atau akses ke data sensitif, hubungi dukungan AWS pelanggan secara langsung.

Dapatkah Asisten AWS Mitra memberikan tanggapan yang dipersonalisasi?

Saat ini, Asisten AWS Mitra hanya menyediakan informasi dan panduan umum. Itu tidak dapat menghasilkan konten kreatif atau melakukan tugas. Jika Asisten AWS Mitra tidak dapat menjawab pertanyaan Anda secara memadai, ajukan tiket dukungan atau kirimkan umpan balik. Berikut ini adalah contoh tambahan:

Lakukan	Jangan
Di mana saya dapat menemukan tanggal perpanjangan tingkat saya?	Kapan tanggal perpanjangan tingkat saya?
Bagaimana cara mendaftar di jalur?	Daftarkan kami di jalur Perangkat Lunak.
Bagaimana cara mendaftar untuk program MSP?	Dapatkah saya mendaftar untuk program MSP?

Note

Untuk pertanyaan terkait akun, mitra harus menggunakan saluran [AWS dukungan](#) yang ada.

Dapatkah AWS mitra menggunakan Asisten Mitra untuk mengubah akun atau layanan mereka?

Tidak, tetapi ini dapat memandu Anda tentang cara membuat perubahan ini atau mengarahkan Anda untuk mendukung tim yang bisa.

Bagaimana Asisten AWS Mitra menangani informasi rahasia?

AWS Asisten Mitra dirancang untuk menjaga privasi dan data pengguna, tetapi kami tidak menyarankan untuk berbagi informasi sensitif.

Apa yang terjadi jika Asisten AWS Mitra tidak memahami pertanyaan saya?

Jika Asisten AWS Mitra tidak dapat menjawab pertanyaan Anda, coba ulangi. Atau, ini dapat memandu Anda ke tim dukungan untuk bantuan. Dalam skenario ini, kirimkan umpan balik jempol ke bawah bersama dengan konteks tambahan, yang akan membantu kami meningkatkan kinerja chatbot. Secara opsional, Anda dapat memberikan umpan balik terstruktur dari daftar pilihan kategori umum.

Apa yang harus saya lakukan jika Asisten AWS Mitra tidak merespons?

Coba segarkan halaman atau mulai ulang aplikasi. Jika masalah berlanjut, laporkan masalahnya ke tim dukungan kami.

Apa yang harus saya lakukan jika Asisten AWS Mitra memberikan informasi yang salah?

Jika Anda yakin Asisten AWS Mitra telah memberi Anda informasi yang salah, berikan umpan balik di akhir sesi Anda menggunakan opsi jempol ke bawah.

Dapatkah saya mematikan Asisten AWS Mitra jika saya tidak ingin menggunakannya?

AWS Asisten Mitra adalah fitur opsional. Jika Anda memilih untuk tidak menggunakannya, hubungi tim dukungan kami.

Mempersiapkan produk Anda untuk AWS Marketplace

Topik berikut menjelaskan bagaimana mempersiapkan produk perangkat lunak Anda untuk dijual AWS Marketplace. Mempersiapkan produk Anda melibatkan konfigurasi paket Anda, menetapkan skema harga, menentukan kategori di mana untuk daftar produk Anda, dan menambahkan kata kunci sehingga produk Anda muncul dalam pencarian yang relevan. Jika Anda adalah vendor perangkat lunak independen (ISV), mitra saluran, penyedia layanan terkelola (MSP), atau individu yang memiliki sesuatu untuk ditawarkan yang bekerja dengan AWS produk dan layanan, Anda dapat menggunakan proses ini untuk mempersiapkan mempublikasikan produk di AWS Marketplace

Topik

- [Pengiriman produk untuk AWS Marketplace](#)
- [Penunjukan untuk produk yang berjalan AWS](#)
- [Harga produk untuk AWS Marketplace](#)
- [Wilayah dan negara untuk AWS Marketplace produk Anda](#)
- [Terjemahan dan bahasa](#)
- [Menggunakan kontrak standar di AWS Marketplace](#)
- [Menyediakan metadata untuk produk AWS Marketplace](#)
- [Membuat AMI dan petunjuk penggunaan produk kontainer untuk AWS Marketplace](#)
- [Mengoptimalkan AWS Marketplace produk Anda untuk pencarian](#)
- [Tingkatkan AWS Marketplace produk Anda dengan media promosi](#)

Pengiriman produk untuk AWS Marketplace

Setiap metode pengiriman produk memiliki beberapa opsi untuk pengemasan, harga, dan pengiriman. Beberapa metode tidak tersedia untuk Anda sebagai penjual AWS Marketplace sampai Anda mendaftar untuk program yang mendukungnya.

Anda dapat membuat produk dengan harga daftar standar dan perjanjian lisensi pengguna akhir (EULA). Anda juga dapat membuat penawaran pribadi untuk pelanggan individu dengan harga khusus dan EULAs. Jika Anda perlu membuat perubahan tambahan pada ketentuan kontrak, Anda dapat bekerja dengan AWS Marketplace tim untuk membuat penawaran pribadi khusus. Untuk menyederhanakan proses pengadaan, Anda dapat menggunakan [persyaratan lisensi standar](#) untuk daftar produk publik dan penawaran pribadi.

Note

AWS menawarkan penjual tertentu opsi untuk memberikan demonstrasi terpandu. AWS Marketplace Jika Anda adalah AWS mitra Jaringan Mitra (APN) yang memenuhi syarat untuk [APN Customer Engagements \(ACE\)](#) dan Anda ingin memberikan opsi ini kepada pembeli, lihat [Menambahkan penawaran pribadi dan tombol permintaan demo](#) untuk informasi selengkapnya.

Tabel berikut mencantumkan metode yang dapat Anda gunakan untuk mengirimkan produk perangkat lunak, dan filter yang digunakan AWS Marketplace pembeli untuk menemukan setiap jenis metode pengiriman di AWS Marketplace konsol.

Metode pengiriman produk	Filter Metode Pengiriman di konsol	Deskripsi
AMI Tunggal	Gambar Mesin Amazon (AMI)	<p>Anda mengirimkan satu Gambar Mesin Amazon kustom (AMI) untuk produk Anda. AMI menyediakan informasi yang diperlukan untuk meluncurkan instans Amazon Elastic Compute Cloud (Amazon EC2).</p> <p>Pembeli dapat menggunakan AMI tunggal untuk membuat EC2 instans Amazon dengan produk Anda yang sudah diinstal dan siap digunakan.</p> <p>Untuk informasi selengkapnya, lihat Produk berbasis AMI di AWS Marketplace.</p>
AMIs dikirimkan menggunakan AWS CloudFormation template	CloudFormation Template	Anda dapat membuat daftar produk berbasis AMI yang dikirimkan ke AWS

Metode pengiriman produk	Filter Metode Pengiriman di konsol	Deskripsi
		<p>Marketplace pembeli dengan menggunakan CloudFormation templat.</p> <p>Untuk informasi selengkapnya tentang pengiriman AMIs sebagai CloudFormation templat, lihat menggunakan pengiriman berbasis AMI. AWS CloudFormation</p> <p>Untuk informasi selengkapnya tentang CloudFormation templat, lihat AWS CloudFormation konsep di Panduan AWS CloudFormation Pengguna.</p>
EC2 Produk berbasis komponen Image Builder	EC2 Komponen Image Builder	<p>Anda dapat membuat daftar produk yang dikirim ke AWS Marketplace pembeli menggunakan komponen EC2 Image Builder.</p> <p>Untuk informasi selengkapnya tentang penggunaan komponen EC2 Image Builder, lihat Bekerja dengan produk komponen EC2 Image Builder di Referensi AWS Marketplace API.</p>

Metode pengiriman produk	Filter Metode Pengiriman di konsol	Deskripsi
Produk atau aplikasi berbasis kontainer	Kontainer	<p>Anda mengirimkan produk yang dikemas dalam gambar kontainer. Produk kontainer terdiri dari opsi, yang merupakan sekumpulan gambar kontainer dan templat penerapan yang bekerja sama.</p> <p>Untuk informasi selengkapnya, lihat Produk berbasis kontainer pada AWS Marketplace.</p>
Produk data	AWS Data Exchange	<p>Anda menggunakan AWS Data Exchange untuk membuat produk data.</p> <p>Untuk informasi tentang memublikasikan dan mengelola produk dan produk data melalui AWS Data Exchange, lihat Menyediakan produk data di AWS Data Exchange di Panduan Pengguna AWS Data Exchange.</p>

Metode pengiriman produk	Filter Metode Pengiriman di konsol	Deskripsi
Algoritma pembelajaran mesin dan paket model	SageMaker Model	<p>Anda menggunakan Amazon SageMaker AI untuk membuat algoritma atau paket model, dan kemudian mempublikasikannya AWS Marketplace.</p> <p>Untuk informasi selengkapnya tentang penyampaian algoritma pembelajaran mesin dan paket model, lihat Produk machine learning.</p> <p>Untuk informasi tentang SageMaker AI, lihat Apa itu SageMaker AI? di Panduan Pengembang SageMaker AI Amazon.</p>
Perangkat lunak sebagai layanan (SaaS)	SaaS	<p>Anda dapat menawarkan produk SaaS dengan berbasis langganan, berbasis kontrak, atau kontrak dengan model harga konsumsi.</p> <p>Untuk informasi selengkapnya, lihat Produk berbasis SaaS di AWS Marketplace.</p>
Layanan profesional	Layanan profesional	<p>Anda dapat menawarkan layanan profesional yang mendukung atau bekerja dengan AWS Marketplace produk lain.</p>

Penunjukan untuk produk yang berjalan AWS

Note

Penunjukan berikut mulai berlaku pada 1 Mei 2025.

Produk yang digunakan pada AWS menerima penunjukan khusus dalam hasil AWS Marketplace pencarian dan halaman detail produk mereka. Tabel berikut mencantumkan jenis pengiriman produk yang dapat menerima penunjukan.

Metode pengiriman produk	Ditunjuk sebagai dikerahkan pada AWS
Berbasis AMI	Ya
Berbasis kontainer	Ya, kecuali untuk produk yang digunakan di Amazon EKS Anywhere. Untuk informasi lebih lanjut tentang produk kontainer yang mengaktifkan fungsi ini, lihat Mengintegrasikan AWS Marketplace untuk Kontainer Di Mana Saja dengan License Manager nanti dalam panduan ini.
SageMaker Algoritma AI dan berbasis paket model	Ya
Berbasis SaaS	Hanya untuk produk yang berjalan sepenuhnya a AWS. Untuk informasi lebih lanjut, lihat Pedoman arsitektur nanti dalam panduan ini.
Berbasis data	Ya
Berbasis layanan profesional	Tidak

Harga produk untuk AWS Marketplace

Sebagai AWS Marketplace penjual, Anda dapat memilih model harga untuk produk Anda. Untuk produk berbayar, AWS Marketplace kumpulkan biaya perangkat lunak dari pelanggan. Semua harga

didasarkan pada dolar AS (USD). Tidak ada biaya layanan untuk perangkat lunak gratis atau sumber terbuka yang tersedia untuk pelanggan tanpa biaya. Topik ini memberikan informasi tentang model penetapan harga dan harga untuk produk perangkat lunak di AWS Marketplace.

Untuk informasi tentang pengembalian uang, lihat [Pengembalian uang produk di AWS Marketplace](#).

Topik

- [Model penetapan harga](#)
- [Mengubah model harga](#)
- [Mengubah harga](#)
- [Penawaran privat](#)
- [Pengembalian uang produk di AWS Marketplace](#)

Model penetapan harga

Topik berikut memberikan informasi umum tentang model harga yang tersedia di AWS Marketplace.

Topik

- [Harga tahunan](#)
- [Harga penggunaan](#)
- [Harga kontrak](#)
- [Bawa harga Lisensi Anda Sendiri](#)

Untuk informasi tentang model harga untuk metode pengiriman produk tertentu, lihat:

- [Harga produk AMI untuk AWS Marketplace](#)
- [Harga produk kontainer untuk AWS Marketplace](#)
- [Harga produk pembelajaran mesin untuk AWS Marketplace](#)
- [Harga produk SaaS di AWS Marketplace](#)
- [Harga produk layanan profesional di AWS Marketplace](#)

Harga tahunan

Model harga tahunan memungkinkan Anda menawarkan produk kepada pelanggan yang dapat membeli langganan 12 bulan. Sebagai contoh, harga berlangganan dapat memberikan penghematan

hingga 40 persen dibandingkan dengan menjalankan produk yang sama setiap jam untuk waktu yang lama. Pelanggan ditagih untuk jumlah penuh kontrak pada saat berlangganan. Untuk informasi selengkapnya tentang bagaimana langganan tahunan disajikan kepada pelanggan, lihat [langganan AMI](#) atau [Model harga untuk produk kontainer berbayar](#).

Pertimbangan saat bekerja dengan langganan tahunan meliputi:

- Harga tahunan ditentukan per jenis instans. Ini bisa sama untuk semua jenis instans Amazon Elastic Compute Cloud (Amazon EC2) atau berbeda untuk setiap jenis instans.
- Semua jenis instans Tahunan juga harus memiliki tipe instance Per Jam yang ditentukan. AWS Marketplace tidak menawarkan harga khusus Tahunan atau Per Jam tanpa Tahunan pada produk yang sama. Untuk setiap penawaran produk Harga tahunan, Harga per jam juga perlu ditentukan.
- Harga tahunan \$0 diperbolehkan pada jenis instans tertentu, jika harga Per Jam juga \$0 dan ada jenis instans Tahunan non-\$0 lainnya yang ditentukan.
- Pada akhir periode berlangganan tahunan, pelanggan akan mulai ditagih dengan harga per jam.
- Jika pelanggan membeli langganan X Tahunan tetapi menjalankan perangkat lunak Y pada instans Y, maka pelanggan dikenakan biaya pada harga perangkat lunak Per Jam untuk instans (Y-X) yang tidak tercakup oleh langganan Tahunan. Dengan demikian, tarif per jam harus disertakan untuk semua jenis instans harga Tahunan.
- Menggunakan penawaran pribadi penjual, Anda dapat menawarkan AMI multi-tahun (hingga 3 tahun) atau durasi khusus dengan pembayaran di muka, atau jadwal pembayaran yang fleksibel. Untuk informasi selengkapnya tentang kontrak multi-tahun dan durasi khusus, lihat [Mempersiapkan penawaran pribadi untuk AWS Marketplace produk Anda](#) dan [Paket cicilan penawaran pribadi](#).

Jika Anda menawarkan produk Tahunan di AWS Marketplace, Anda menyetujui kebijakan pengembalian dana khusus untuk produk Tahunan, yang terletak di bagian dokumen Pengunggah File di [Portal Manajemen AWS Marketplace](#).

Perubahan harga

Anda dapat mengubah harga tahunan (nilai \$, misalnya \$1.000/tahun menjadi \$1.200/tahun) setiap 90 hari. Namun, Anda harus memberikan pemberitahuan 90 hari kepada pelanggan yang sudah ada tentang harga tahunan. Harga baru akan berlaku untuk langganan baru tetapi tidak akan berdampak pada langganan yang ada.

Note

Selama periode pemberitahuan 90 hari, Anda tidak dapat memperbarui jenis instans yang didukung.

Perubahan harga akan berlaku untuk perpanjangan otomatis hanya jika harga diubah setidaknya 90 hari sebelum tanggal perpanjangan otomatis. Pelanggan akan menerima pesan email sebelum perpanjangan otomatis yang mencakup harga baru.

Perjanjian lisensi pengguna akhir

Penggunaan perangkat lunak oleh AWS pelanggan selama 12 bulan di bawah langganan tahunan dicakup oleh EULA yang Anda berikan di halaman AWS Marketplace detail produk Anda.

Harga penggunaan

Model penetapan harga penggunaan, juga dikenal sebagai harga bayar saat Anda pergi, memungkinkan Anda untuk menawarkan produk kepada pelanggan yang hanya membayar untuk apa yang mereka gunakan.

Sebagai penjual, Anda dapat memilih salah satu dari kategori penggunaan berikut:

- Pengguna
- Tuan rumah
- Bandwidth
- Data
- Tingkatan
- Unit (untuk kategori kustom)

Anda juga dapat menentukan hingga 24 dimensi untuk produk. Biaya diukur dan dilaporkan ketika API dipanggil oleh perangkat lunak. Kami menyarankan agar penjual mengonfigurasi API untuk dipanggil sekali per jam sebagai praktik terbaik, tergantung pada kasus penggunaannya. Semua penggunaan dihitung setiap bulan dan ditagih setiap bulan menggunakan mekanisme yang sama dengan AWS Marketplace perangkat lunak yang ada.

Menggunakan AWS Marketplace Metering Service, Anda dapat menangani beberapa skenario harga baru.

Example Biaya oleh Host

Jika perangkat lunak Anda memantau host, Anda dapat mengenakan biaya untuk setiap host yang dipantau dan menetapkan harga yang berbeda berdasarkan ukuran host.

Example Biaya oleh Pengguna

Jika perangkat lunak Anda memungkinkan beberapa pengguna di seluruh organisasi, Anda dapat mengenakan biaya berdasarkan pengguna. Setiap jam, pelanggan dikenakan biaya untuk jumlah total pengguna yang ditetapkan.

Note

Dalam Formulir Beban Produk (PLF), kolom yang relevan didahului dengan “FCP” (Harga Konsumsi Fleksibel). Misalnya: Kategori FCP (Kategori Harga Kustom).

Untuk AWS Marketplace Metering Service produk, perhatikan hal berikut:

- Jika perangkat lunak Anda sudah AWS Marketplace aktif, Anda perlu membuat produk untuk mengaktifkan dimensi penggunaan alternatif. Anda tidak dapat mengonversi produk standar untuk menggunakan AWS Marketplace Metering Service. Setelah produk baru diterbitkan, Anda dapat menghapus produk lama atau menyimpan keduanya di situs web.
- AWS Marketplace Metering Service mengharuskan perangkat lunak Anda melaporkan penggunaan setiap jam, mencatat penggunaan pelanggan selama satu jam. Jika ada kegagalan dalam transmisi atau penerimaan catatan layanan pengukuran, tidak AWS akan dapat menagih untuk penggunaan tersebut. Anda bertanggung jawab untuk memastikan keberhasilan penerimaan catatan pengukuran.
- Produk yang menggunakan AWS Marketplace Metering Service tidak mendukung 1-Klik. Pembeli diminta untuk meluncurkan perangkat lunak Anda dengan peran AWS Identity and Access Management (IAM) dengan izin khusus dan memiliki gateway internet.
- Uji Coba Gratis dan Harga Tahunan tidak kompatibel dengan AWS Marketplace Metering Service.
- Mengubah dimensi (pengguna, host, bandwidth, dan data) atau nama dimensi tidak didukung. Anda perlu membuat produk baru.

Harga kontrak

Dengan menggunakan model penetapan harga kontrak, Anda dapat memberikan harga di muka kepada pelanggan yang memungkinkan mereka membeli lisensi selama 1 bulan, 12 bulan, 24 bulan, atau 36 bulan.

Harga kontrak tersedia untuk produk-produk berikut:

- Produk berbasis AMI tunggal dan AMI dengan produk berbasis AWS CloudFormation template. Untuk informasi selengkapnya, lihat [Harga kontrak untuk produk AMI pada AWS Marketplace](#)
- Produk berbasis kontainer. Untuk informasi selengkapnya, lihat [Harga kontrak untuk produk kontainer](#).
- Perangkat lunak sebagai produk berbasis layanan (SaaS). Untuk informasi selengkapnya, lihat [Harga untuk kontrak SaaS](#).

Note

Harga kontrak untuk AMI dan produk berbasis kontainer hanya untuk produk baru. Jika Anda memiliki AMI atau produk berbasis kontainer yang sudah ada dan ingin menggunakan harga kontrak, buat daftar baru lalu terapkan model penetapan harga kontrak dengan menggunakan Formulir Pemuatan Produk (PLF) untuk menambahkan dimensi yang berbeda, mengintegrasikan AMI atau produk berbasis kontainer dengan, AWS License Manager lalu publikasikan AMI atau produk berbasis kontainer. Ketika pelanggan membeli produk dengan harga kontrak, lisensi dibuat oleh AWS Marketplace pelanggan Akun AWS bahwa perangkat lunak Anda dapat memeriksa menggunakan License Manager API. Pelanggan akan membutuhkan peran IAM untuk meluncurkan instance AMI atau produk berbasis kontainer.

Bawa harga Lisensi Anda Sendiri

Tidak ada biaya layanan untuk produk Bring Your Own License (BYOL). AWS Marketplace

Untuk memenuhi janji pilihan pelanggan kami, kami mengharuskan semua produk BYOL juga memiliki opsi berbayar. Ini agar pelanggan yang tidak memiliki lisensi yang ada memiliki opsi untuk membeli dan menggunakan produk.

Untuk produk BYOL, kami menyadari bahwa pembelian perangkat lunak secara online adalah penyimpangan dari bagaimana beberapa perusahaan melakukan bisnis. Oleh karena itu, selama 90 hari pertama setelah peluncuran, kami akan melonggarkan persyaratan bahwa perangkat lunak ini disertai dengan versi yang tersedia untuk pembelian AWS Marketplace. Selama waktu ini, tim manajemen AWS Marketplace akan bekerja dengan Anda untuk mengatasi tantangan. Tim dapat membantu Anda menentukan apakah dan bagaimana perangkat lunak dapat dibuat tersedia untuk dibeli AWS Marketplace.

Mengubah model harga

Perubahan model harga harus ditinjau dan disetujui oleh AWS Marketplace untuk memastikan pengalaman pelanggan yang positif dan mengurangi risiko bagi semua pihak. Diskusikan perubahan model harga yang ingin Anda lakukan dengan menghubungi tim [Operasi AWS Marketplace Penjual](#).

Semua permintaan untuk perubahan model harga dapat memakan waktu 30-90 hari untuk diproses dan ditinjau.

Note

Harap dicatat bahwa perubahan model harga tidak didukung untuk produk SaaS. Untuk informasi lebih lanjut tentang harga produk SaaS, lihat harga produk [SaaS](#) di AWS Marketplace

Mengubah harga

Anda dapat memperbarui harga dan metadata melalui Portal Manajemen AWS Marketplace

Untuk mengubah harga

1. Masuk ke [Portal Manajemen AWS Marketplace](#).
2. Di tab Produk, daftar produk saat ini yang Anda buat tersedia. Edit daftar produk Anda atau minta perubahan di sini.

Note

Untuk pelanggan baru, perubahan harga segera berlaku. Untuk pelanggan penawaran umum yang ada, perubahan harga berlaku pada hari pertama bulan setelah periode 90 hari.

Misalnya, jika Anda mengirim pemberitahuan perubahan harga pada 16 Maret, perubahan harga terjadi pada hari pertama bulan yang mengikuti periode 90 hari, yaitu 1 Juli.

Penawaran privat

Dalam program Penawaran Pribadi AWS Marketplace Penjual, AWS Marketplace penjual dapat menegosiasikan harga khusus dan EULAs dengan AWS Marketplace pelanggan individu (pembeli). Lihat informasi yang lebih lengkap di [Mempersiapkan penawaran pribadi untuk AWS Marketplace produk Anda](#).

Pengembalian uang produk di AWS Marketplace

Note

Jika Anda bertransaksi melalui operator Amazon Web Services EMEA, hanya pengguna terverifikasi yang dapat memproses pengembalian dana, mengubah informasi Know Your Customer (KYC), dan mengubah informasi keuangan seperti detail rekening bank. Pengguna terverifikasi adalah pengguna terverifikasi KYC dan pengguna sekunder yang menyelesaikan verifikasi. Untuk informasi lebih lanjut tentang proses KYC, lihat [Selesaikan proses Mengenal Pelanggan Anda](#).

Semua produk berbayar AWS Marketplace, terlepas dari model harga, harus memiliki kebijakan pengembalian dana yang dinyatakan untuk biaya perangkat lunak. Kebijakan pengembalian dana harus mencakup ketentuan pengembalian dana, dan metode menghubungi penjual untuk meminta pengembalian dana.

Sebagai penjual, rincian kebijakan pengembalian dana terserah Anda. Namun, kami mendorong Anda untuk menawarkan pembeli beberapa cara pengembalian dana untuk penggunaan produk. Anda harus mematuhi kebijakan pengembalian dana yang Anda posting. Topik ini memberikan informasi tentang jenis permintaan pengembalian dana AWS Marketplace produk, kebijakan terkait dan proses persetujuan, dan bagaimana Anda dapat mengirimkan permintaan pengembalian dana untuk pembeli.

Topik

- [Jenis permintaan pengembalian dana untuk AWS Marketplace produk](#)
- [AWS Marketplace kebijakan pengembalian produk dan persetujuan](#)

- [Meminta pengembalian dana produk](#)
- [Meminta pengembalian dana massal](#)

Jenis permintaan pengembalian dana untuk AWS Marketplace produk

Pembeli dapat meminta berbagai jenis pengembalian uang untuk AWS Marketplace produk. Untuk AWS Marketplace produk yang dijual oleh AWS, lihat halaman kebijakan pengembalian dana dan kemudian kirimkan formulir dukungan kontak menggunakan AWS Support Center Console. Jika pembeli meminta pengembalian dana perangkat lunak langsung dari AWS, kami menginstruksikan mereka untuk menghubungi penjual menggunakan informasi kontak dukungan Anda yang diposting untuk produk yang dimaksud. Pengembalian biaya AWS infrastruktur apa pun tergantung pada kebijaksanaan AWS dan ditangani secara independen dari pengembalian dana perangkat lunak.

Untuk produk yang dijual oleh pihak ketiga, pembeli harus melihat kebijakan pengembalian dana pada halaman detail produk. Biaya perangkat lunak untuk AWS Marketplace langganan dibayarkan kepada penjual produk, dan pengembalian uang harus diminta dari penjual secara langsung. Setiap AWS Marketplace penjual harus menyertakan kebijakan pengembalian dana di AWS Marketplace halaman mereka.

AWS Marketplace kebijakan pengembalian produk dan persetujuan

Daftar berikut menjelaskan kebijakan AWS Marketplace pengembalian dana dan apakah persetujuan Anda diperlukan:

- Uji coba gratis

Jika Anda mencantumkan perangkat lunak Anda sebagai produk uji coba gratis, AWS dapat mengeluarkan pengembalian uang atas nama Anda untuk biaya perangkat lunak yang timbul dalam waktu tujuh hari sejak konversi dari uji coba gratis ke langganan berbayar. Pengembalian uang yang dikeluarkan sehubungan dengan konversi uji coba gratis tidak memerlukan tindakan dari pihak Anda. Dengan mengaktifkan uji coba gratis pada suatu produk, Anda menyetujui kebijakan ini.

- Penawaran pribadi

Semua pengembalian uang untuk penawaran pribadi harus diotorisasi oleh Anda sebelum AWS dapat memprosesnya.

- Pengembalian uang pengukuran perangkat lunak

Jika Anda menggunakan AWS Marketplace Metering Service untuk mengukur penggunaan perangkat lunak Anda, AWS dapat mengeluarkan pengembalian uang atas nama Anda untuk biaya perangkat lunak yang dihasilkan dari kesalahan pengukuran perangkat lunak. Jika kesalahan ini umum terjadi di beberapa pembeli, AWS berhak untuk menentukan pengembalian dana yang sesuai untuk setiap pembeli dan menerapkannya langsung ke setiap pembeli. Pengembalian uang yang dikeluarkan sehubungan dengan AWS Marketplace Metering Service harus dikonfirmasi dengan penjual satu kali, tetapi tidak mengharuskan penjual untuk mengkonfirmasi setiap pengembalian dana individu. Dengan menggunakan produk, Anda menyetujui kebijakan ini. AWS Marketplace Metering Service

- Pembatalan langganan dalam waktu 48 jam setelah pembelian

Jika pembeli membatalkan langganan mereka dalam waktu 48 jam setelah pembelian penawaran non-pribadi, AWS mengeluarkan pengembalian dana penuh (batalan dengan pengembalian dana 100 persen). Pengembalian uang yang dikeluarkan sehubungan dengan pembatalan dalam waktu 48 jam setelah pembelian tidak memerlukan tindakan dari pihak Anda. Setelah 48 jam, permintaan pembeli tersebut sesuai dengan kebijaksanaan Anda. Dengan mencantumkan produk Anda AWS Marketplace, Anda menyetujui kebijakan ini.

- Upgrade langganan

Jika pembeli mengganti langganan penawaran non-pribadi yang ada dengan langganan yang lebih mahal atau langganan dengan nilai yang sama, AWS dapat mengeluarkan pengembalian uang atas nama Anda untuk langganan tingkat bawah. Ini adalah proses dua langkah untuk pembeli: Beli langganan baru dan kemudian minta pembatalan langganan lama dengan pengembalian dana.

- Downgrade langganan

Semua permintaan pengembalian dana langganan downgrade harus diotorisasi oleh Anda sebelum AWS dapat memprosesnya.

Semua pengembalian dana AWS resmi diproses secara otomatis dan tidak memerlukan tindakan dari pihak Anda.

Meminta pengembalian dana produk

Gunakan prosedur berikut untuk meminta pengembalian dana perangkat lunak untuk pembeli eksternal atau akun pengujian internal.

Untuk meminta pengembalian dana perangkat lunak

1. Kumpulkan informasi berikut:

- Akun AWS ID Pembeli (12 digit) yang digunakan untuk berlangganan produk. Jika pembeli adalah pembayar organisasi mereka, dapatkan Akun AWS ID dari akun tertaut yang digunakan untuk berlangganan.
- Alamat email pembeli yang terkait dengan mereka Akun AWS.
- Akun AWS ID Penjual (12 digit).
- ID Produk. Anda dapat menemukan ID Produk di dasbor [perjanjian dan perpanjangan atau dasbor pendapatan](#) yang [ditagih](#).
- ID Penawaran. Anda dapat menemukan ID Penawaran di dasbor [perjanjian dan perpanjangan atau dasbor pendapatan yang ditagih](#), tab Penawaran, atau tab Perjanjian di [Portal Manajemen AWS Marketplace](#)
- Periode penagihan.
- Nomor faktur dari dasbor [pendapatan yang ditagih](#).
- Jumlah pengembalian dana sebelum pajak.

2. Masuk ke [Portal Manajemen AWS Marketplace](#).

3. Pilih Support.

4. Pilih Minta Pengembalian Dana.

5. Masukkan Akun AWS nomor pelanggan, ID produk, dan periode penagihan.

6. Di kolom Komentar tambahan (opsional), masukkan deskripsi singkat tentang permintaan, termasuk ID produk, jenis pengembalian dana, dan nomor faktur (jika tersedia). Misalnya:

Cancel the buyer's subscription to *<product-ID>* and *<offer-ID>* and process a full refund for invoice *<invoice-number>*.

7. Untuk mengajukan permintaan pengembalian dana, peningkatan, atau penurunan versi langganan tahunan, lakukan hal berikut:

- a. Verifikasi bahwa pembeli telah membeli langganan tahunan menggunakan dasbor [perjanjian dan perpanjangan atau dasbor pendapatan](#) yang [ditagih](#).
- b. Di kolom Komentar tambahan (opsional), masukkan tanggal pembatalan langganan dan tentukan apakah pembeli meminta pengembalian dana, peningkatan, atau penurunan versi.

8. Kirim formulir. Kami akan diberitahu dan akan mulai memproses pengembalian dana dan menerbitkannya kepada pembeli.

9. Untuk mengelola permintaan pengembalian dana, tim dukungan AWS Marketplace pembeli akan membuat kasus dukungan yang sesuai di [Konsol Dukungan Tengah](#). Status pengembalian dana akan muncul di baris subjek kasus dukungan. Status dapat berupa salah satu dari yang berikut:
 - Selesai — Pengembalian dana diproses dan tidak diperlukan tindakan lebih lanjut.
 - Tertunda — Pengembalian dana akan diproses setelah siklus penagihan saat ini berakhir.
 - Tindakan yang Diperlukan — Permintaan tidak dapat diproses, dan kami memerlukan informasi tambahan dari Anda. Anda dapat menanggapi langsung kasus dukungan; Namun, Anda juga perlu mengirimkan formulir permintaan pengembalian dana baru.

Pengembalian dana akan muncul di akun pembeli dalam waktu 24-48 jam. Namun, dibutuhkan waktu hingga lima hari kerja agar dana muncul di rekening keuangan pembeli.

Meminta pengembalian dana massal

Langkah-langkah berikut menjelaskan cara membuat permintaan pengembalian dana yang melibatkan 20+ faktur atau 20+ akun, Proses ini merampingkan permintaan pengembalian dana skala besar dan memastikan bahwa Anda memberikan informasi yang diperlukan.

Untuk meminta pengembalian dana massal

1. Kumpulkan informasi yang diperlukan berikut:
 - Akun AWS ID penjual (12 digit)
 - Daftar semua pembeli Akun AWS IDs (masing-masing 12 digit). Anda dapat memasukkan beberapa pembeli, atau pembeli tunggal untuk beberapa periode penagihan.
 - Semua produk IDs
 - Penawaran IDs jika berlaku
 - Semua faktur IDs
 - Periode penagihan untuk setiap faktur
 - ID AWS Marketplace Referensi Tiket Pengembalian Dana
2. Kompilasi informasi yang diperlukan ke dalam spreadsheet CSV dengan kolom berikut:
 - **Seller Account ID**
 - **Subscriber Account ID**
 - **Payer Account ID**

- **Billing Period**
 - **Invoice ID**
 - **Targeted Amount**
 - **Product ID**
3. Masuk ke [Konsol Dukungan Tengah](#) sebagai pengguna root.

 Note

Anda tidak dapat menyelesaikan langkah-langkah ini kecuali Anda masuk sebagai pengguna root.

4. Buat kasus dukungan ke Akun dan penagihan dan pilih AWS Marketplace.
5. Di baris subjek, masukkan **Request Bulk Refund**.
6. Di bawah Komentar Tambahan, masukkan yang berikut ini:
 - Jumlah total akun yang terpengaruh
 - Jumlah total faktur
 - Jumlah pengembalian dana total
 - Pembeneran bisnis untuk pengembalian dana
 - AWS Marketplace Referensi ID Referensi Tiket Pengembalian
7. Lampirkan file CSV dan buat kasus dukungan.
8. Kasus dukungan dialihkan ke agen berikutnya yang tersedia untuk bantuan.

Wilayah dan negara untuk AWS Marketplace produk Anda

Saat Anda membuat produk AWS Marketplace, Anda memilih AWS Wilayah di mana produk tersebut tersedia. Anda juga memilih negara tempat pembeli dapat membeli produk Anda. Kedua properti ini serupa, tetapi tidak sama. Misalnya, pembeli mungkin berlokasi di, dan membeli dari, Amerika Serikat, tetapi mereka mungkin berencana untuk menginstal produk Anda di Wilayah Eropa (Frankfurt). Agar pembeli ini dapat membeli produk Anda, Anda harus memasukkan Amerika Serikat dalam daftar negara Anda, dan Wilayah Eropa (Frankfurt) dalam daftar Wilayah Anda. Anda dapat menggunakan topik ini untuk mempelajari lebih lanjut tentang Wilayah dan negara untuk AWS Marketplace produk Anda.

AWS Daerah

Saat membuat atau mengedit informasi produk server atau pembelajaran mesin, Anda dapat membatasi produk Anda ke AWS Wilayah tertentu tempat pengguna Anda dapat menginstal dan menggunakan produk.

Untuk produk server, termasuk Amazon Machine Image (AMI) -, container-, dan produk AWS CloudFormation berbasis, Anda dapat memilih Wilayah tertentu di mana produk tersedia. Anda juga dapat memilih untuk secara otomatis membuat produk Anda tersedia di Wilayah AS baru, Wilayah non-AS, atau semua Wilayah saat tersedia.

Untuk produk pembelajaran mesin, Anda dapat memilih Wilayah tertentu, atau semua Wilayah termasuk Wilayah future saat tersedia.

Untuk informasi selengkapnya tentang AWS Wilayah, lihat [titik akhir AWS layanan](#) di Referensi AWS Umum.

Negara-negara

Secara default, produk Anda tersedia untuk pembeli di semua negara AWS Marketplace yang tersedia. Untuk produk server dan perangkat lunak sebagai layanan (SaaS) yang baru dan yang sudah ada, Anda dapat mengontrol ketersediaan produk di negara tertentu untuk tujuan pajak, kepatuhan, dukungan, atau pemasaran.

Ada pengecualian untuk fungsi ini:

- Pembelian sebelumnya — Setelah memperbarui produk Anda dengan daftar negara baru, pembeli yang telah berlangganan produk Anda akan tetap memiliki akses saat langganan mereka aktif.
- Penawaran pribadi — Ketika Anda membatasi produk Anda untuk pembeli di negara tertentu, itu tidak membatasi penawaran pribadi. Ketika Anda membuat penawaran pribadi kepada pembeli tertentu, itu tersedia untuk pembeli itu, bahkan jika mereka berada di negara yang tidak Anda sertakan di negara yang Anda tentukan.

Note

Kelayakan pelanggan ditentukan pada tingkat akun AWS tertaut. Untuk informasi selengkapnya, lihat [Bagaimana cara AWS menentukan Lokasi akun Anda?](#)

Pelanggan yang berbagi hak mereka hanya dapat mengaktifkan hak di wilayah yang Anda izinkan. Untuk informasi selengkapnya tentang mengelola hak, lihat [Berbagi langganan di organisasi di Panduan Pembeli](#).AWS Marketplace

Terjemahan dan bahasa

Secara default, AWS Marketplace menerjemahkan daftar dan penawaran produk Anda ke dalam bahasa berikut:

- Perancis (fr-fr)
- Jepang (Ja-jp)
- Korea (KO-kr)
- Spanyol (ES-es)

AWS Marketplace juga menyediakan satu set contoh kontrak standar, untuk referensi saja, dalam bahasa-bahasa tersebut. Kontrak tidak mengikat secara hukum. Untuk informasi selengkapnya tentang kontrak, lihat [bahasa yang didukung](#) di Panduan AWS Marketplace Pembeli.

Topik berikut menjelaskan cara kerja terjemahan, cara mengunduh kontrak standar yang diterjemahkan, dan cara memilih keluar dari terjemahan.

Topik

- [Memulai](#)
- [Memilih keluar dari terjemahan](#)

Memulai

Ingat hal berikut tentang AWS Marketplace terjemahan:

- Untuk membantu memperluas jangkauan Anda ke pasar global, AWS Marketplace terjemahkan konten bahasa Inggris yang Anda berikan tanpa upaya tambahan dari Anda. Jika Anda memiliki masalah dengan terjemahan, gunakan halaman [Hubungi Kami](#) di Portal AWS Marketplace Manajemen. Pilih kategori Daftar Produk dan Bahasa Lokal Opt-out, dan berikan umpan balik tentang masalah ini.

AWS Marketplace dapat membuat perubahan terbatas, seperti tidak menerjemahkan istilah teknis atau nama produk.

- Anda dapat memilih untuk tidak menerjemahkan daftar Anda. Jika Anda melakukannya, pembeli melihat konten Anda dalam bahasa Inggris dan AWS Marketplace konten terkait apa pun dalam bahasa pilihan mereka. Untuk informasi lebih lanjut tentang memilih keluar, lihat [Memilih keluar dari terjemahan](#) di bawah ini.
- Anda harus mengirimkan pendaftaran penjual dan informasi daftar produk Anda dalam bahasa Inggris.
- Anda dapat mengirimkan EULAs dalam bahasa yang didukung. Daftar produk terkait harus ditargetkan secara geografis, dan bahasa harus menjadi bahasa utama negara tersebut.

AWS Marketplace menyediakan satu set kontrak contoh standar, hanya untuk referensi, diterjemahkan ke dalam bahasa yang tercantum di atas. Anda dapat mengunduh kontrak dari [bahasa yang Didukung](#) di Panduan AWS Marketplace Pembeli.

- Sebagai praktik terbaik, ketika Anda mempublikasikan produk baru, tinjau terjemahan dengan produk dalam keadaan terbatas. [Hubungi Kami](#) dengan masalah apa pun sebelum Anda mempublikasikan produk.
- Proses penerjemahan dimulai setelah Anda memindahkan produk ke keadaan terbatas. Pembeli melihat teks bahasa Inggris sampai terjemahannya tersedia.

Memilih keluar dari terjemahan

Langkah-langkah berikut menjelaskan cara memilih untuk tidak menerjemahkan daftar produk Anda. Anda dapat memilih keluar dari beberapa atau semua bahasa. Misalnya, Anda dapat memiliki daftar yang diterjemahkan hanya ke dalam bahasa Prancis dan Jepang.

Biarkan 3 - 5 hari agar opt-out berlaku.

Untuk memilih keluar

1. Buka [Portal AWS Marketplace Manajemen](#) dan pilih Hubungi Kami.
2. Pilih kategori berikut dalam urutan yang tercantum:
 - Marketplace Komersil
 - Daftar Produk
 - Bahasa Lokal Menyisih

3. Sertakan ID daftar dan bahasa yang ingin Anda tolak.
4. Sebagai praktik terbaik, bantu pembeli Anda memahami mengapa mereka melihat teks bahasa Inggris. Misalnya:
 - a. Saat memilih keluar dari semua terjemahan, tambahkan **English** ke akhir nama produk Anda.
 - b. Saat memilih keluar dari satu atau lebih terjemahan, tambahkan beberapa teks penjelasan di akhir deskripsi produk Anda. Sebagai contoh: **We have declined automatic translation services for this product.**
5. Kirim permintaan Anda.

Menggunakan kontrak standar di AWS Marketplace

Saat Anda [menyiapkan produk Anda](#) sebagai AWS Marketplace penjual, Anda perlu menentukan perjanjian lisensi pengguna akhir (EULA) mana yang akan mengatur penggunaan produk Anda. Anda dapat menggunakan yang berikut ini:

- EULA Anda sendiri.
- Standard Contract for AWS Marketplace (SCMP), template kontrak yang dapat membantu merampingkan alur kerja pengadaan dan mempercepat transaksi. Anda juga dapat menggunakan Federal Addendum untuk SCMP untuk membantu mempercepat pengadaan Pemerintah AS.
- Kontrak Reseller untuk AWS Marketplace (RCMP), templat kontrak pengecer standar yang dapat digunakan. ISVs

Topik berikut menjelaskan cara menggunakan templat AWS Marketplace kontrak standar.

Topik

- [Kontrak Standar untuk AWS Marketplace](#)
- [Kontrak Reseller untuk AWS Marketplace](#)

Kontrak Standar untuk AWS Marketplace

AWS Marketplace mengembangkan [Kontrak Standar untuk AWS Marketplace \(SCMP\)](#) bekerja sama dengan komunitas pembeli dan penjual. SCMP mengatur penggunaan dan mendefinisikan kewajiban pembeli dan penjual untuk solusi digital. Contoh solusi digital termasuk perangkat lunak server,

perangkat lunak sebagai layanan (SaaS), dan algoritma kecerdasan buatan dan pembelajaran mesin (AI/ML)).

SCMP secara proaktif mendefinisikan kesamaan di seluruh klausul kontrak utama seperti penggunaan, garansi, ganti rugi, dan hukum yang mengatur. Penjual dapat menawarkan persyaratan SCMP sebagai EULA untuk transaksi swalayan, di mana pembeli dapat mencari, membeli, dan menyebarkan solusi dengan cepat. Untuk [penawaran pribadi](#), pembeli dapat meminta template SCMP dari penjual, dan persyaratan dapat diubah untuk memenuhi persyaratan transaksi khusus sebagaimana disepakati oleh para pihak.

Anda juga dapat menggunakan addendum opsional berikut dengan SCMP untuk penawaran swalayan atau pribadi:

- [Addendum Keamanan yang Ditingkatkan](#) — Mendukung transaksi dengan persyaratan keamanan data yang tinggi.
- [Addendum Asosiasi Bisnis HIPAA](#) — Mendukung transaksi dengan persyaratan kepatuhan Undang-Undang Portabilitas dan Akuntabilitas Asuransi Kesehatan 1996 (HIPAA).
- [Addendum Federal](#) — Mendukung pembelian perangkat lunak yang melibatkan Pemerintah AS.

Sanggahan

EULA adalah antara Anda dan pembeli. Menggunakan SCMP sebagai EULA Anda adalah kebijaksanaan Anda. Dengan menerapkan SCMP ke daftar produk Anda, Anda memilih program SCMP. Di bawah program ini, AWS dapat memperbarui template SCMP secara berkala dan dapat memperbarui daftar produk yang membawa persyaratan dengan versi saat ini. Anda dapat menarik diri dari program SCMP kapan saja dengan mengganti template SCMP dengan EULA Anda sendiri.

Memulai dengan SCMP

Anda dapat memperbarui EULA ke SCMP dan menawarkannya kepada pembeli produk Amazon Machine Instance (AMI) dan perangkat lunak sebagai layanan (SaaS) tunggal yang baru dan yang sudah ada. Prosedur yang Anda gunakan untuk meminta pembaruan tergantung pada apakah suatu produk terdaftar atau tidak melalui daftar layanan mandiri (SSL).

Jika Anda memiliki pertanyaan, kirim pesan email ke tim Kontrak AWS Marketplace Standar di aws-mp-standardcontract@amazon.com.

Untuk memperbarui EULA ke SCMP untuk produk AMI dan SaaS yang terdaftar melalui SSL

1. Tinjau ketentuan [Kontrak Standar untuk AWS Marketplace](#).
2. Masuk ke [Portal Manajemen AWS Marketplace](#).
3. Untuk produk, pilih jenis produk dan pilih daftar produk yang ingin Anda perbarui kontraknya.
4. Pilih Minta Perubahan, lalu pilih Perbarui wilayah dan harga.
5. Pilih Kontrak Standar AWS Marketplace untuk EULA jika belum dipilih.
6. Di Catatan & Pemberitahuan, ketik **Please update this product to SCMP V2**.
7. Tinjau perubahan, dan pilih Kirim untuk ditinjau.

Untuk memperbarui EULA ke SCMP untuk produk AMI dan SaaS yang tidak terdaftar melalui SSL

1. Buka halaman [Hubungi kami](#) di Portal Manajemen AWS Marketplace.
2. Masuk ke akun AWS Marketplace penjual Anda.
3. Lengkapi formulir sebagai berikut:
 - Untuk subjek pertanyaan Anda, pilih Commercial Marketplace.
 - Untuk kategori, pilih Daftar Produk.
 - Untuk subkategori, pilih Permintaan Kontrak Standar.
 - Dalam deskripsi, ketik **Please update these products to SCMP V2** dan berikan judul produk dan IDs yang ingin Anda perbarui dengan SCMP.

Kontrak Reseller untuk AWS Marketplace

Kontrak Reseller untuk AWS Marketplace (RCMP) adalah templat kontrak pengecer standar yang ISVs dapat digunakan saat mengotorisasi mitra saluran untuk menjual kembali produk ISV kepada pembeli. AWS Marketplace Kontrak membantu mengurangi redundansi dalam tinjauan kontrak hukum dan mempercepat waktu ke pasar ketika ISVs dan mitra saluran masuk ke dalam hubungan pengecer dan/atau menggunakan persyaratan pengecer untuk penawaran [pribadi mitra saluran](#) mereka (CPPO). Saat membuat CPPO, ISVs dapat mengunggah kontrak ke peluang (Otorisasi Penjualan Kembali), dan kemudian mitra saluran dapat melihat dan menerima kontrak. AWS Marketplace pembeli tidak dapat melihat RCMP.

Sanggahan

RCMP adalah kontrak opsional untuk ISVs. Jika ISVs memutuskan untuk menjual kembali produk mereka melalui mitra saluran, mereka dapat melampirkan RCMP atau persyaratan kontrak khusus mereka sendiri — yang sudah ada atau yang sudah dinegosiasikan sebelumnya — saat menciptakan peluang.

Memulai dengan RCMP

Bagian ini menjelaskan cara meninjau istilah dan menggunakan RCMP.

Untuk menggunakan RCMP sambil menciptakan peluang AWS Marketplace

1. Tinjau ketentuan [Kontrak Reseller untuk AWS Marketplace](#).
2. Ikuti langkah-langkah di [Buat peluang jual kembali untuk mitra saluran](#) sambil mengacu pada panduan [RCMP](#).

Menyediakan metadata untuk produk AWS Marketplace

Sebagai penjual, saat Anda menambahkan produk AWS Marketplace, Anda menentukan metadata produk. Metadata produk mencakup nama, deskripsi, kategori, dan kata kunci yang Anda gunakan untuk menggambarkan produk Anda bagi pelanggan. AWS Marketplace merevisi metadata produk semata-mata untuk jaminan kualitas dan koreksi kesalahan. Topik ini memberikan informasi dan praktik terbaik yang dapat Anda, sebagai AWS Marketplace penjual, gunakan untuk memberi nama, mendeskripsikan, dan mengkategorikan produk Anda secara efektif.

Note

Semua tab dan bagian halaman detail produk, seperti Harga, Penggunaan, Produk serupa, atau perbandingan Produk, bersifat dinamis dan dikelola oleh AWS Marketplace. Penjual tidak dapat menyesuaikan halaman detail produk untuk menambah atau menghapus bagian.

Penamaan dan mendeskripsikan produk Anda

Informasi yang Anda berikan tentang produk Anda dapat dilihat oleh pembeli. Pastikan bahwa pembeli potensial memiliki informasi yang cukup untuk membuat keputusan tentang membeli produk Anda.

Membuat nama produk

Pertimbangkan panduan berikut saat Anda membuat nama produk:

- Gunakan huruf judul (menulis huruf pertama dari setiap kata penting dalam huruf besar)
- Pastikan bahwa pembeli dapat mengidentifikasi produk dengan nama saja
- Gunakan nama merek atau pabrikan
- Hindari data deskriptif atau hiperbola

Contoh nama produk: Smart Solution Load Balancer - Premium Edition.

Karakter yang didukung

Karakter yang didukung untuk bidang metadata adalah:

- Kode karakter ASCII 0 hingga 126
 - Untuk informasi lebih lanjut lihat Tabel [ASCII Standar](#).
- Hak Cipta ©
- Terdaftar®
- Merek Dagang™
- Simbol mata uang ¢ £ ₪ ¥

Menulis deskripsi produk

Deskripsi produk mencantumkan fitur, manfaat, dan penggunaan produk. Hal ini juga dapat memberikan informasi produk lain yang relevan dan spesifik. Deskripsi dapat memiliki panjang hingga 350 karakter.

Pertimbangkan panduan berikut saat Anda menulis deskripsi produk:

- Hindari penggunaan huruf besar yang tidak perlu

- Hindari tanda baca yang tidak perlu
- Jangan sertakan informasi pengalihan
- Periksa ejaan dan tata bahasa
- Sertakan hanya informasi penting dan berguna
- Hindari data deskriptif dan hiperbola

Contoh solusi produk: Solusi Cerdas secara otomatis mendistribusikan lalu lintas aplikasi yang masuk di beberapa instans Amazon EC2 . Solusi ini mengizinkan Anda untuk mencapai toleransi kesalahan yang lebih besar dalam aplikasi Anda, memberikan jumlah kapasitas penyeimbangan beban yang Anda butuhkan untuk merespons kemudian lintas aplikasi yang masuk. Smart Solution mendeteksi instans yang tidak sehat di kumpulan dan secara otomatis mengubah rute kemudian lintas ke status sehat hingga instans yang tidak sehat dipulihkan. Anda dapat mengaktifkan Smart Solution dalam satu AWS Availability Zone atau di beberapa Availability Zone untuk memastikan performa aplikasi yang lebih konsisten.

Menulis sorotan produk

Halaman informasi produk menampilkan hingga tiga poin sorotan produk. Gunakan poin ini untuk secara singkat menguraikan titik penjualan utama produk.

Contoh sorotan produk: Memproyeksikan biaya: Dengan Smart Solution, Anda hanya membayar atas apa yang Anda gunakan. Anda akan dikenai biaya untuk setiap jam atau sebagian jam saat Smart Solution berjalan.

Menulis informasi dukungan

Pelanggan harus dapat dengan mudah mendapatkan bantuan dengan masalah, seperti menggunakan layanan, pemecahan masalah, dan meminta pengembalian uang (jika ada). Anda harus menentukan opsi kontak dukungan seperti email, nomor telepon, atau tautan formulir web dukungan di halaman arahan pemenuhan.

Memilih kategori dan kata kunci

Saat Anda mencantumkan produk, Anda dapat memilih hingga tiga kategori perangkat lunak dan subkategori yang sesuai untuk produk Anda. Ini membantu pembeli menemukan produk Anda saat mereka menelusuri atau mencari produk AWS Marketplace. Pilih hanya kategori yang relevan dengan produk Anda; dalam kebanyakan kasus, hanya satu kategori yang berlaku. Formulir beban produk dan tab Produk berisi daftar lengkap kategori.

Kategori tidak sama dengan kata kunci. Kategori dan subkategori yang tersedia telah ditentukan sebelumnya AWS Marketplace, dan Anda memutuskan mana yang berlaku untuk produk Anda dengan memilihnya dari daftar selama proses permintaan produk. Kata kunci tidak ditentukan sebelumnya, tetapi dibuat selama proses. Anda tidak perlu menambahkan kategori sebagai kata kunci.

Membuat kata kunci pencarian

Selama proses permintaan produk, Anda dapat memasukkan hingga tiga kata kunci (satu kata atau frasa) untuk membantu pembeli menemukan produk Anda melalui pencarian di situs. Bidang kata kunci dapat berisi maksimal 250 karakter.

Kiat berikut dapat membantu Anda membuat kumpulan kata kunci pencarian yang relevan:

- Gunakan istilah yang relevan.
- Jangan menggunakan nama produk yang dipublikasikan oleh penjual lain atau menggunakan nama penjual lain.
- Pilih kata kunci dari kosakata pembeli Anda—yaitu, kata-kata dan frasa yang mungkin digunakan pembeli saat memikirkan tipe produk Anda.
- Buat kata kunci berdasarkan fitur tertentu dalam produk Anda.
- Jangan gunakan judul produk sebagai kata kunci. Judul produk sudah diindeks dalam pencarian.

Note

Kata kunci tidak sama dengan kategori perangkat lunak. Kata kunci adalah istilah yang lebih spesifik yang terkait dengan produk Anda.

Membuat AMI dan petunjuk penggunaan produk kontainer untuk AWS Marketplace

Sebagai AWS Marketplace penjual, Anda mengirimkan produk Anda ke pembeli dengan cara yang berbeda. Saat mengirimkan dengan Amazon Machine Images (AMIs) dan gambar kontainer, Anda bertanggung jawab untuk menulis instruksi penggunaan untuk mereka. Bagian berikut memberi Anda persyaratan dan rekomendasi tentang menulis petunjuk penggunaan untuk jenis produk ini.

Untuk informasi selengkapnya, lihat [Produk berbasis AMI di AWS Marketplace](#).

Topik

- [Persyaratan](#)
- [Menulis catatan rilis](#)
- [Menulis instruksi penggunaan](#)
- [Menulis instruksi peningkatan](#)
- [Menulis instruksi CloudFormation pengiriman](#)

Persyaratan

Saat membuat petunjuk penggunaan untuk produk Anda, Anda harus menyertakan informasi berikut:

- Lokasi semua informasi sensitif yang disimpan oleh pelanggan.
- Jelaskan semua konfigurasi enkripsi data. Berikan instruksi terperinci tentang bagaimana pengguna berinteraksi dengan aplikasi Anda untuk mendekripsi data yang diperlukan jika aplikasi Anda menggunakan teknik enkripsi apa pun.
- Jika produk Anda menyertakan materi kriptografi, Anda harus menyertakan persyaratan rotasi dalam instruksi penggunaan. Lihat persyaratan dasar [the section called “Persyaratan produk berbasis AMI”](#) untuk daftar yang menggunakan kredensi dan kunci kriptografi.
- Jika salah satu penyimpanan data dalam produk Anda adalah hak milik, berikan step-by-step instruksi untuk konfigurasi, pencadangan, dan pemulihan.
- Step-by-step petunjuk tentang cara menilai dan memantau kesehatan dan fungsi aplikasi yang tepat. Misalnya:
 - Arahkan ke [EC2 konsol Amazon](#) Anda dan verifikasi bahwa Anda berada di wilayah yang benar.
 - Pilih Instance dan pilih instans yang Anda luncurkan.
 - Pilih server untuk menampilkan halaman metadata Anda dan pilih tab Pemeriksaan status di bagian bawah halaman untuk meninjau apakah pemeriksaan status Anda lulus atau gagal.
- Panduan preskriptif dalam mengelola kuota AWS layanan. Untuk informasi selengkapnya, lihat [Panduan Referensi Umum AWS](#).
- Rincian harga termasuk biaya menjalankan sumber AWS daya yang ditambahkan di atas kuota standar. Ini dapat dimasukkan dalam petunjuk penggunaan produk Anda atau ditautkan ke [dokumentasi](#) yang berisi informasi rinci tentang mengelola dan meminta peningkatan kuota layanan.

Menulis catatan rilis

Setiap kali Anda memperbarui produk, Anda harus memberikan deskripsi tentang perubahan dalam catatan rilis. Catatan rilis harus berisi informasi khusus untuk membantu pengguna memutuskan apakah akan menginstal pembaruan. Gunakan label yang jelas untuk pembaruan, seperti “Kritis” untuk pembaruan keamanan atau “Penting” atau “Opsional” untuk tipe pembaruan lainnya.

Menulis instruksi penggunaan

Berikan petunjuk penggunaan yang membantu memastikan bahwa pembeli dapat berhasil mengonfigurasi dan menjalankan perangkat lunak. Instruksi penggunaan yang Anda berikan ditampilkan selama proses konfigurasi.

Untuk menulis petunjuk penggunaan yang efektif, ikuti panduan berikut:

- Asumsikan pengguna tertarik tetapi tidak mendapat informasi.
- Berikan pengguna semua informasi yang diperlukan untuk meluncurkan dan menggunakan produk Anda, termasuk pengaturan konfigurasi dan langkah-langkah khusus apa pun.

Contoh petunjuk penggunaan:

1. Luncurkan produk menggunakan 1-Klik.
2. Gunakan browser web untuk mengakses aplikasi di `https://<EC2_Instance_Public_DNS>/index.html`.
3. Masuk menggunakan kredensial berikut:
 - Nama pengguna: `user`
 - Kata sandi: contoh ID (`instance_id`)

Menulis instruksi peningkatan

Mbucketikan detail tentang bagaimana pembeli dapat memutakhirkan dari versi sebelumnya dari produk. Sertakan informasi tentang cara menyimpan data dan pengaturan saat membuat instans lain. Jika tidak ada jalur pemutakhiran edit bidang ini untuk secara khusus menyebutkannya.

Contoh petunjuk pemutakhiran:

1. Lakukan ****, kemudian ****.

2. Periksa bahwa semua plugin yang digunakan oleh proyek Anda kompatibel dengan versi *.* , dengan melakukan ***. Jika mereka tidak kompatibel, lakukan ***.
3. Buat cadangan data Anda, dengan melakukan ***.

Menulis instruksi CloudFormation pengiriman

Saat menggunakan CloudFormation pengiriman, Anda juga harus menyertakan yang berikut:

- Tujuan untuk setiap peran AWS Identity and Access Management (IAM) dan kebijakan IAM yang dibuat oleh template AWS CloudFormation
- Tujuan dan lokasi setiap kunci yang dibuat oleh AWS CloudFormation template
- Detail konfigurasi jaringan dalam penerapan yang melibatkan lebih dari satu elemen
- Panduan terperinci tentang bagaimana aplikasi Anda diluncurkan dan bagaimana aplikasi tersebut dikonfigurasi untuk berkomunikasi jika penerapan mencakup beberapa sumber daya AWS
- Rincian harga yang mencakup biaya menjalankan sumber AWS daya yang ditambahkan di atas batas standar. Memberikan panduan preskriptif tentang mengelola batas AWS layanan.
- Semua konfigurasi enkripsi data. Misalnya: enkripsi sisi server Amazon S3, enkripsi Amazon Elastic Block Store (Amazon EBS), Linux Unified Key Setup (LUKS), dll.)

Mengoptimalkan AWS Marketplace produk Anda untuk pencarian

Sebagai AWS Marketplace penjual, Anda dapat mengoptimalkan produk Anda untuk pencarian sehingga pembeli dapat lebih mudah menemukannya. Pencarian adalah alat penting dalam perjalanan pembeli yang memungkinkan pelanggan menemukan produk terbaik untuk memenuhi kebutuhan unik mereka. Untuk AWS Marketplace pelanggan, pencarian terjadi di dua lokasi utama: mesin pencari (misalnya, Google atau Bing) dan fungsi pencarian di AWS Marketplace. Topik ini memberikan informasi tentang cara mengoptimalkan iklan Anda untuk kedua lokasi.

Topik

- [Optimisasi mesin pencari](#)
- [AWS Marketplace pencarian](#)

Optimisasi mesin pencari

Mengoptimalkan halaman detail produk Anda untuk peringkat lebih tinggi untuk kata kunci yang relevan sangat penting untuk mendorong lebih banyak pengunjung unik ke halaman detail produk Anda melalui mesin pencari.

Ada tiga elemen halaman utama yang mendasar dan berdampak untuk meningkatkan pencarian organik ke halaman web apa pun, termasuk halaman detail produk: kata kunci, tag judul, dan tag judul H1.

Kata kunci

Kata kunci adalah elemen sentral untuk optimasi mesin pencari, karena mereka menyaring topik menjadi kueri terfokus yang mendorong hasil mesin pencari. Proses mengidentifikasi kata kunci yang paling relevan untuk halaman Anda melibatkan penelitian kata kunci. Alat optimisasi mesin pencari dapat memberikan informasi berharga, seperti volume pencarian kata kunci (berapa kali per bulan kata kunci dicari di Google), peringkat saat ini, tren pencarian, daya saing kata kunci, dan kata kunci terkait. Dari penelitian ini, Anda dapat mengidentifikasi kata kunci primer dan sekunder.

Kata kunci optimasi mesin pencari utama Anda harus menjadi satu kata atau frasa unik yang mewakili topik utama halaman Anda. Kata kunci utama ini harus secara alami ditenun ke dalam salinan judul produk Anda, deskripsi singkat, dan bagian sorotan. Kata kunci sekunder harus merupakan istilah yang sangat relevan yang ditemukan dalam konten halaman yang tersisa.

Tag judul

Tag judul, yang muncul di halaman hasil mesin pencari dan sebagai judul halaman di jendela atau tab browser, menginformasikan pembaca dan perayap web mesin pencari tentang konten halaman. Untuk halaman detail AWS Marketplace produk, judul produk berfungsi sebagai tag judul, jadi penting untuk mengoptimalkan judul produk Anda dengan kata kunci optimasi mesin pencari untuk meningkatkan potensi peringkat. Untuk meningkatkan kemungkinan mencapai peringkat tinggi di halaman hasil mesin pencari, sertakan nama merek Anda, nama produk, dan kata kunci yang relevan dalam tag judul Anda.

Tag judul H1

Tag heading H1 memiliki tiga peran:

- Mereka membantu pengunjung memindai konten halaman untuk informasi yang mereka butuhkan.

- Mereka meningkatkan aksesibilitas bagi pengunjung dengan gangguan penglihatan yang menggunakan pembaca layar untuk memahami konten halaman.
- Mereka menyediakan kata kunci yang ditemukan di judul halaman, yang menerima bobot relevansi optimasi mesin pencari tambahan jika didukung oleh konten halaman yang mengikuti.

AWS Marketplace pencarian

AWS Marketplace Situs web ini memberi peringkat hasil permintaan pencarian menggunakan teknik pengoptimalan pencarian yang mirip dengan yang digunakan di seluruh industri. Dengan memahami bagaimana AWS Marketplace peringkat dan mengembalikan hasil pencarian, Anda dapat membuat detail produk yang dioptimalkan untuk mesin AWS Marketplace pencari. Sebaiknya pertimbangkan panduan ini saat Anda membuat halaman detail produk Anda.

Kata kunci

Selama proses pembuatan produk, Anda dapat mengirimkan hingga tiga kata kunci (kata atau frasa tunggal) untuk membantu pelanggan menemukan produk Anda melalui pencarian situs. Kotak teks kata kunci dapat berisi hingga 250 karakter.

Gunakan tips berikut untuk membuat kata kunci pencarian:

- Gunakan istilah yang relevan sehingga pelanggan dapat dengan mudah menemukan produk Anda.
- Pilih kata kunci dari kosakata pelanggan Anda—yaitu, kata dan frasa yang cenderung mereka gunakan saat memikirkan jenis produk Anda.
- Buat kata kunci berdasarkan fitur tertentu dalam produk Anda.
- Jangan sertakan judul produk dalam ketentuan yang Anda kirimkan. Judul produk sudah diindeks dalam pencarian.

Note

Kata kunci tidak sama dengan kategori perangkat lunak. Kata kunci adalah istilah yang lebih spesifik yang terkait dengan produk Anda.

Anda dapat mengedit kata kunci setelah Anda membuat produk dengan mengedit metadata untuk produk. Untuk produk yang Anda buat menggunakan tab Produk Portal Manajemen AWS

Marketplace, Anda juga menggunakan tab Produk untuk membuat perubahan. Untuk informasi selengkapnya, lihat [Perubahan dan pembaruan produk](#).

Tim Operasi AWS Marketplace Penjual membantu mengarahkan kueri dengan kata atau kata yang terdengar serupa dengan arti yang serupa. Misalnya, ketika pelanggan mencari mobil ketika Anda mengharapkan mereka untuk mencari mobil.

Kategori perangkat lunak

Saat Anda mencantumkan produk, Anda dapat memilih hingga tiga kategori perangkat lunak dan subkategori yang sesuai untuk produk Anda. Ini membantu pelanggan menemukan produk Anda saat mereka menelusuri atau mencari produk AWS Marketplace. Pilih hanya kategori yang relevan dengan produk Anda. Dalam kebanyakan kasus, hanya satu kategori yang berlaku. Baik formulir pemuatan produk dan halaman Produk berisi daftar lengkap kategori.

Note

Kategori tidak sama dengan kata kunci. Kategori dan subkategori yang tersedia telah ditentukan sebelumnya untuk AWS Marketplace Anda memutuskan mana yang berlaku untuk produk Anda dengan memilihnya dari daftar. Kata kunci tidak ditentukan sebelumnya, tetapi mereka dibuat selama proses.

Bagian Sorotan

Halaman detail produk menampilkan hingga tiga sorotan produk sebagai bullet point. Pelanggan dapat mencari produk berdasarkan sorotan, jadi sertakan sorotan saat Anda membuat produk. Sorotan harus menggambarkan nilai jual utama produk dalam bahasa yang singkat dan informatif.

Example Sorotan

- Biaya proyeksi: Dengan AnyCompany produk, Anda hanya membayar untuk apa yang Anda gunakan. Anda dikenakan biaya untuk setiap jam atau sebagian jam yang sedang berjalan.

Deskripsi singkat

Deskripsi produk mencantumkan fitur, manfaat, dan petunjuk penggunaan produk, bersama dengan informasi produk lain yang relevan dan spesifik. Ingatlah pedoman berikut saat Anda membuat deskripsi produk:

- Hindari kapitalisasi dan tanda baca yang tidak perlu
- Jangan sertakan informasi pengalihan
- Periksa ejaan dan tata bahasa
- Sertakan hanya informasi penting dan berguna

Example Deskripsi singkat

AnyCompanyproduk secara otomatis mendistribusikan lalu lintas aplikasi yang masuk di beberapa instans Amazon EC2 . Hal ini memungkinkan Anda untuk meningkatkan toleransi kesalahan dalam aplikasi Anda dengan mulus menyediakan kapasitas load balancing yang Anda butuhkan untuk menanggapi lalu lintas aplikasi yang masuk. AnyCompanyProduk mendeteksi kejadian yang tidak sehat di kolam dan secara otomatis mengalihkan lalu lintas ke contoh yang sehat sampai kasus yang tidak sehat dipulihkan. Pelanggan dapat mengaktifkannya dalam satu AWS Availability Zone atau di beberapa Availability Zone untuk memungkinkan kinerja aplikasi yang lebih konsisten.

Tingkatkan AWS Marketplace produk Anda dengan media promosi

Sebagai AWS Marketplace penjual, Anda dapat membantu pelanggan menemukan dan mengevaluasi produk Anda dengan lebih baik dengan meningkatkan daftar produk Anda dengan media promosi. Media promosi adalah video dan gambar yang ditampilkan secara mencolok di halaman produk Anda dan memberi pelanggan cara mudah untuk belajar tentang produk Anda. Bagian berikut memberikan praktik terbaik untuk media promosi dan tips untuk menambah dan mengelola media promosi di AWS Marketplace Management Portal (AMMP).

Topik

- [Praktik terbaik untuk media promosi](#)
- [Kiat untuk menambah dan mengelola media promosi](#)

Praktik terbaik untuk media promosi

Hingga lima video dan 10 gambar didukung. Setiap item media promosi harus menyertakan judul yang diperlukan dan deskripsi opsional. Deskripsi digunakan sebagai teks alt untuk media dan sangat disarankan untuk meningkatkan aksesibilitas visual dan optimasi mesin pencari.

Note

Anda harus memiliki hak dan izin yang sesuai untuk mengunggah atau menambahkan media promosi apa pun. Media yang ditambahkan ke produk Anda akan tersedia untuk umum untuk semua pengguna yang menjelajah AWS Marketplace.

Video

Jenis aset ini memberi Anda kesempatan untuk memperkenalkan produk dan perusahaan Anda melalui ikhtisar singkat. Anda juga dapat menyertakan konten seperti wawancara pelanggan, kutipan, manfaat yang relevan, dan titik data. Selain itu, video adalah cara yang bagus untuk menampilkan produk Anda melalui demo atau walk-through yang direkam, terutama untuk fitur utama atau kasus penggunaan. Kami merekomendasikan video berdurasi 2-5 menit, karena video yang lebih pendek lebih berdampak.

Spesifikasi:

- Kami mendukung pengunggahan langsung video melalui AMMP atau melalui akses publik S3 tautan.
- Resolusi video yang disarankan adalah 1080p (1920x1080 piksel) dengan rasio aspek 16:9 yang disukai.
- Format video harus.mp4 dengan ukuran file maksimal 50MB.
- (Opsional) Gambar sampul dapat ditambahkan sebagai pengganti penutup yang dibuat secara otomatis.
 - Gambar sampul harus memiliki resolusi 500x281 piksel (rasio aspek 16:9).
 - Format gambar harus.png (lebih disukai), .jpg, atau.svg tanpa transparansi.

Note

Video yang dihosting secara eksternal tidak didukung secara langsung. Publik S3 tautan akan di-cache AWS Marketplace selama penambahan dan setiap perubahan pasca-penambahan tidak akan tercermin sampai media ditambahkan lagi.

YouTube video didukung dengan kemampuan terbatas:

- Jika bidang URL video produk (bidang terpisah dari media promosi) menyertakan tautan langsung ke YouTube video, video akan langsung disematkan sebagai item media promosi terakhir jika penyematan diaktifkan untuk video.
- Hanya satu URL video produk yang didukung (yaitu hanya satu YouTube video) dan pemesanan ulang media yang YouTube disematkan tidak didukung.

Gambar

Jenis aset ini memberi Anda kesempatan untuk mempromosikan produk Anda melalui tangkapan layar dan panggilan keluar konsep utama. Gambar ideal untuk menjelaskan fitur produk yang kompleks melalui diagram ringkas atau diagram alir. Selain itu, gambar dapat digunakan untuk menjelaskan tingkatan harga dan pengelompokan yang tidak dapat ditangani dengan deskripsi dimensi harga.

Spesifikasi:

- Kami mendukung pengunggahan langsung gambar melalui AMMP atau melalui akses publik S3 tautan.
- Resolusi gambar yang disarankan adalah 780x439 piksel dengan rasio aspek 16:9 yang disukai. Maksimum 3480x3480 piksel.
- Format gambar harus .png (lebih disukai), .jpg, atau .svg tanpa transparansi.
- Thumbnail dibuat secara otomatis dari gambar sumber asli.

Kiat untuk menambah dan mengelola media promosi

Tambahkan media

- Media promosi dapat ditambahkan dengan login ke [AWS Marketplace Management Portal](#) dan membuat produk baru atau memperbarui produk yang sudah ada. Bagian Gambar dan video digunakan untuk menambahkan media promosi dan terletak di dalam area informasi Produk.
- Sementara API Bisa juga digunakan untuk menambah media promosi, upload langsung hanya didukung melalui AMMP. Publik S3 tautan didukung di keduanya API and AMMP.
- Tambahkan media promosi sesuai urutan yang Anda inginkan untuk ditampilkan oleh media. Item pertama akan menjadi media unggulan di halaman produk.
- Saat jenis aset gambar digunakan, gambar sampul opsional tidak tersedia. Gambar akan memiliki thumbnail yang dibuat secara otomatis setelah ditambahkan.

Perbarui atau hapus media

- Setelah media promosi ditambahkan ke suatu produk, judul, deskripsi, dan sampul (jika tersedia) dapat disesuaikan sesuai kebutuhan dengan mengedit produk.
- Media promosi dapat dihapus setelah ditambahkan dengan mengklik Hapus untuk aset tersebut di AMMP sebelum pengiriman.

Memesan media

- Secara default, aset media promosi baru ditambahkan ke akhir pesanan aset.
- Untuk memesan ulang media, hapus aset yang rusak dan tambahkan lagi dalam urutan yang tepat untuk pengiriman.

Mempersiapkan penawaran pribadi untuk AWS Marketplace produk Anda

Penawaran pribadi adalah istilah yang dinegosiasikan yang digunakan untuk membeli produk dari AWS Marketplace. Ini dapat melibatkan rencana harga khusus, perjanjian lisensi pengguna akhir (EULA), atau solusi khusus. Penjual dan pembeli bernegosiasi sebelum melakukan penawaran pribadi yang berbeda dari penawaran umum. Anda dapat membuat dan memperluas beberapa penawaran pribadi ke satu pembeli. Pembeli yang Anda memperpanjang penawaran pribadi untuk memiliki opsi untuk memilih antara penawaran pribadi dan penawaran umum. Pembeli hanya dapat berlangganan satu penawaran pada waktu tertentu. Mereka tidak dapat berlangganan penawaran pribadi dan penawaran umum pada saat yang bersamaan. Topik ini memberikan informasi tentang cara kerja penawaran pribadi, termasuk pertimbangan khusus, pengalaman pembeli, dan laporan penjual.

Note

AWS menawarkan pembeli dengan kasus penggunaan unik atau perusahaan AWS Marketplace untuk meminta penawaran pribadi untuk produk Anda langsung dari halaman detail produk. Jika Anda adalah AWS mitra Jaringan Mitra (APN) yang memenuhi syarat untuk [APN Customer Engagements \(ACE\)](#) dan Anda ingin memberikan opsi ini kepada pembeli, lihat [Menambahkan penawaran pribadi dan tombol permintaan demo](#) untuk informasi selengkapnya.

Topik

- [Bagaimana penawaran pribadi bekerja](#)
- [Pertimbangan penawaran pribadi](#)
- [Pengalaman penawaran pribadi untuk pembeli](#)
- [Pelaporan untuk penawaran pribadi](#)
- [Jenis produk yang didukung untuk penawaran AWS Marketplace pribadi](#)
- [Membuat dan mengelola penawaran pribadi](#)
- [Membuat penawaran pribadi sebagai Mitra AWS Marketplace Saluran](#)
- [Paket cicilan penawaran pribadi](#)

- [Membuat perjanjian bertanggal future untuk penawaran pribadi](#)
- [FAQ penawaran pribadi](#)

Bagaimana penawaran pribadi bekerja

Anda dapat membuat dan mengelola penawaran pribadi Anda dari halaman Penawaran di [Portal Manajemen AWS Marketplace](#). Anda menentukan produk untuk penawaran untuk menghasilkan ID dan URL unik. Anda akan membuat rencana harga untuk penawaran pribadi, menambahkan persyaratan hukum dan dokumen penjualan, dan memperluas penawaran ke pembeli tertentu Akun AWS. Penawaran ini hanya terlihat oleh akun tempat Anda membuat penawaran.

Setelah Anda membuat penawaran pribadi dan memberi tahu pembeli potensial, mereka dapat melihat dan menerima penawaran tersebut. Untuk melihat penawaran, pembeli harus masuk ke Akun AWS yang menerima penawaran.

Note

Pembeli tidak dapat melihat penawaran kecuali Anda memperluasnya ke akun tertaut atau akun manajemen mereka. Anda tidak dapat memberikan batasan layanan dalam penawaran, sehingga pembeli dapat menggunakan sebanyak mungkin produk Anda dengan harga yang dinegosiasikan seperti yang mereka inginkan, kecuali produk tersebut memiliki batas.

Untuk informasi tentang membuat penawaran pribadi, lihat [Membuat dan mengelola penawaran pribadi](#).

Penawaran pribadi dilacak dalam laporan penjual. Untuk informasi selengkapnya, lihat [Melaporkan penawaran pribadi](#) dan [panduan laporan Penjual](#).

Pertimbangan penawaran pribadi

Saat bekerja dengan penawaran pribadi, pertimbangkan hal berikut:

- Saat Anda menambahkan dukungan untuk jenis instans baru atau Wilayah AWS, pelanggan yang sudah berlangganan penawaran pribadi untuk produk Anda tidak akan dapat mengakses instans atau Wilayah yang baru ditambahkan secara otomatis. Anda harus membuat penawaran pribadi lain dengan instans dan Wilayah yang ingin diakses pelanggan. Setelah menerima penawaran

baru, pelanggan dapat mengakses instans dan Wilayah yang baru ditambahkan. Pelanggan yang berlangganan produk Anda di masa mendatang juga dapat mengaksesnya, selama mereka termasuk dalam penawaran pribadi. Untuk informasi selengkapnya tentang cara membuat penawaran pribadi baru, lihat [Mengubah perjanjian di AWS Marketplace](#).

- Anda tidak dapat membuat penawaran pribadi untuk pihak kedua, Amazon Machine Image (AMI) bulanan, atau multi-AMI-based pengiriman menggunakan AWS CloudFormation produk, atau untuk membatasi penggunaan pelanggan.
- Untuk penawaran pribadi dengan paket angsuran, dimungkinkan untuk membagi pembayaran di muka menjadi beberapa pembayaran dari waktu ke waktu. Untuk informasi selengkapnya, lihat [Paket cicilan penawaran pribadi](#).
- Jika akun pembeli untuk penawaran pribadi Anda dikelola melalui pasar pribadi, Anda harus menyertakan akun pembeli dan akun yang menyertakan administrator pasar pribadi mereka dalam penawaran.
- Penawaran pribadi tidak mendukung model Bring Your Own License (BYOL).
- Gunakan opsi EULA Kustom saat membuat penawaran pribadi dengan ketentuan kontrak negosiasi unik dalam penawaran pribadi Anda. Anda dapat melampirkan hingga lima dokumen.
- Untuk kontrak perangkat lunak sebagai layanan (SaaS) dan kontrak SaaS dengan produk konsumsi, Anda dapat menawarkan peningkatan dan pembaruan pada perjanjian yang dibuat ketika pembeli menerima penawaran pribadi. Misalnya, Anda dapat melakukan ini untuk memberikan hak baru, menawarkan diskon harga, menyesuaikan jadwal pembayaran, atau mengubah perjanjian lisensi pengguna akhir (EULA) untuk menggunakan persyaratan lisensi standar. Untuk informasi selengkapnya, lihat [Mengubah perjanjian di AWS Marketplace](#).

Pengalaman penawaran pribadi untuk pembeli

Saat pembeli menavigasi ke halaman langganan produk Anda, spanduk menunjukkan bahwa penawaran pribadi tersedia. Setelah pembeli menerima penawaran, mereka ditagih untuk pembelian menggunakan alat portal yang sama yang digunakan untuk semua AWS Marketplace transaksi. Penawaran yang diterima menjadi perjanjian. Pembeli dapat menemukan detail perjanjian di bagian Kelola Langganan AWS Management Console, dan penjual dapat menemukan detailnya di tab Perjanjian. Portal Manajemen AWS Marketplace

AWS Marketplace pembeli dapat mengakses pembiayaan pihak ketiga untuk penawaran pribadi. Untuk informasi lebih lanjut, lihat [Pembiayaan pelanggan sekarang tersedia di AWS Marketplace](#).

Note

Penawaran hanya dapat diterima sebelum tanggal kedaluwarsa. Jika penawaran kedaluwarsa, penawaran akan dipindahkan ke tab Penawaran yang Diterima dan kedaluwarsa.

Untuk melihat dan menerima penawaran pribadi	Pembeli bisa
Dari AWS Marketplace konsol	<p>Arahkan ke Penawaran pribadi di AWS Marketplace konsol dan pilih ID penawaran dari tab Penawaran yang tersedia.</p> <p>Untuk informasi selengkapnya tentang pengalaman pembeli untuk penawaran pribadi, lihat Penawaran pribadi di Panduan AWS Marketplace Pembeli.</p>
Menggunakan tautan yang disediakan penjual	<p>Ikuti tautan yang dikirim oleh penjual untuk langsung mengakses penawaran pribadi.</p> <p>Untuk informasi selengkapnya, lihat Mengirim penawaran pribadi ke pembeli.</p>
Dari halaman produk Anda	<p>Arahkan ke halaman produk untuk produk, dan pilih tautan di spanduk untuk melihat penawaran pribadi.</p> <p>Untuk informasi selengkapnya tentang pengalaman pembeli untuk penawaran pribadi, lihat Penawaran pribadi di Panduan AWS Marketplace Pembeli.</p>

Pelaporan untuk penawaran pribadi

Penawaran pribadi muncul di laporan penjual yang ada dan dalam laporan yang relevan dengan penawaran. [Laporan pendapatan bulanan yang ditagih](#) Ini dihasilkan setiap bulan dan menawarkan visibilitas dan menawarkan informasi ID. Ketika faktur dibuat untuk pembeli, itu muncul dalam laporan yang mencakup periode penagihan yang sesuai. Untuk informasi selengkapnya, lihat [Dasbor penjual](#).

Bidang ID Penawaran berisi ID penawaran unik yang dihasilkan untuk penawaran pribadi. Ini kosong kecuali entri laporan untuk penawaran pribadi. Bidang Visibilitas Penawaran menunjukkan apakah entri laporan adalah penawaran publik atau pribadi. Untuk semua penawaran pribadi, entri ditandai pribadi.

Jenis produk yang didukung untuk penawaran AWS Marketplace pribadi

Sebagai AWS Marketplace penjual, Anda dapat menggunakan penawaran pribadi untuk menjual jenis produk berikut: Amazon Machine Images (AMIs), kontainer, layanan profesional, pembelajaran mesin (ML), dan perangkat lunak sebagai layanan (SaaS) produk. Penawaran pribadi adalah istilah yang dinegosiasikan yang digunakan untuk membeli produk dari AWS Marketplace. Topik ini memberikan informasi tentang penawaran pribadi untuk AMI, kontainer, SaaS, ML, dan produk layanan profesional.

Untuk informasi selengkapnya tentang penawaran privat, lihat [Mempersiapkan penawaran pribadi untuk AWS Marketplace produk Anda](#).

Topik

- [Penawaran pribadi untuk produk AMI](#)
- [Penawaran pribadi untuk produk kontainer](#)
- [Penawaran pribadi untuk produk layanan profesional](#)
- [Penawaran pribadi untuk produk SaaS](#)
- [Penawaran pribadi untuk produk ML](#)

Penawaran pribadi untuk produk AMI

Anda dapat memberikan harga penawaran pribadi untuk produk AMI.

Penawaran dapat berupa durasi khusus untuk hal-hal berikut:

- AMI per jam atau AMI per jam dengan penawaran pribadi tahunan: hingga 3 tahun (1.095 hari). Hanya AMI per jam dengan penawaran pribadi tahunan atau kontrak AMI yang mendukung penjadwalan pembayaran yang fleksibel.
- Penawaran pribadi kontrak AMI: hingga 5 tahun (60 bulan)

Untuk kontrak AMI, penawaran pribadi tidak memantau penggunaan.

Pembeli dapat secara manual meningkatkan ke tingkat kontrak baru kapan saja. Namun, terserah vendor perangkat lunak independen (ISV) untuk menentukan tingkatan kontrak, menegakkan batasan layanan, dan menyarankan pembeli untuk secara manual meningkatkan kontrak mereka dengan lebih banyak unit. Hanya kontrak berbasis harga non-tier yang mendukung peningkatan saat ini. Durasi kontrak penawaran pribadi dapat cocok dengan daftar produk publik, atau dapat berupa durasi khusus dalam beberapa bulan (hingga 60).

Hak lisensi dimulai pada tanggal pembeli menerima penawaran pribadi.

Untuk penawaran pribadi AMI dengan jadwal pembayaran yang fleksibel, Anda dapat mengatur jumlah jenis instans tahunan yang disepakati dalam kontrak, selama kontrak berlangsung.

Note

Penawaran pribadi tidak tersedia untuk kontrak penagihan bulanan.

Penawaran pribadi untuk produk kontainer

Anda dapat memberikan harga penawaran pribadi untuk kontrak produk berbasis kontainer.

Penawaran dapat berupa durasi khusus untuk hal-hal berikut:

- Kontainer per jam atau kontainer per jam dengan penawaran pribadi jangka panjang - Hingga 3 tahun (1.095 hari). Hanya kontainer per jam dengan penawaran pribadi jangka panjang atau kontrak kontainer yang mendukung penjadwalan pembayaran yang fleksibel.
- Penawaran pribadi kontrak kontainer - Hingga 5 tahun (60 bulan)

Untuk kontrak Kontainer, penawaran pribadi tidak memantau penggunaan. Upgrade untuk kontrak kontainer hanya dimungkinkan jika Anda menggunakan harga yang tidak berjenjang.

Pembeli dapat secara manual meningkatkan ke tingkat kontrak baru kapan saja. Namun, vendor perangkat lunak independen (ISV) mendefinisikan tingkatan kontrak, memberlakukan batasan layanan, dan menyarankan pembeli untuk secara manual meningkatkan kontrak mereka dengan lebih banyak unit. Hanya kontrak berbasis harga non-tier yang mendukung peningkatan saat ini. Durasi kontrak penawaran pribadi dapat cocok dengan daftar produk publik, atau dapat berupa durasi khusus dalam beberapa bulan (hingga 60 bulan).

Hak lisensi dimulai pada tanggal pembeli menerima penawaran pribadi. Untuk penawaran pribadi kontainer dengan jadwal pembayaran yang fleksibel, Anda dapat mengatur jumlah unit yang disepakati dalam kontrak, selama kontrak berlangsung. Anda juga dapat menentukan harga per jam khusus untuk unit yang sama jika pembeli menggunakan lebih banyak.

Note

Penawaran pribadi tidak tersedia untuk kontrak penagihan bulanan.

Penawaran pribadi untuk produk layanan profesional

Semua penawaran produk layanan profesional dilakukan melalui penawaran pribadi. Untuk informasi selengkapnya, lihat [Buat penawaran pribadi](#).

Penawaran pribadi untuk produk SaaS

Produk penawaran pribadi perangkat lunak sebagai layanan (SaaS) tidak dapat mengubah tingkat harga untuk tingkat harga tertentu berdasarkan waktu. Misalnya, penawaran tidak dapat membebankan biaya \$0,80/jam selama tiga bulan dan kemudian mengubah harga menjadi \$0,60/jam setelahnya untuk tingkat harga yang sama. Untuk kontrak SaaS, penawaran pribadi tidak memantau penggunaan.

Pembeli dapat secara manual meningkatkan ke tingkat kontrak baru kapan saja. Namun, vendor perangkat lunak independen (ISV) mendefinisikan tingkatan kontrak, memberlakukan batasan layanan, dan menyarankan pembeli untuk secara manual meningkatkan ke tingkatan kontrak yang lebih tinggi bila diperlukan. Durasi kontrak penawaran pribadi dapat cocok dengan daftar produk publik, atau dapat berupa durasi khusus dalam beberapa bulan (hingga 60 bulan).

Penawaran pribadi untuk produk ML

Produk penawaran pribadi Machine Learning (ML) memberi pembeli tertentu harga yang berbeda dari harga yang ditampilkan secara publik. Seperangkat persyaratan dan perjanjian antara Anda dan pembeli dalam penawaran pribadi dapat berbeda dari yang ada di penawaran umum atau penawaran pribadi lainnya.

Penawaran pribadi bekerja dengan salah satu dari beberapa cara:

- Per Jam — Penawaran pribadi dapat berupa tarif per jam yang berbeda dari tarif per jam yang ditampilkan secara publik. Tarif per jam ini abadi karena penawaran pribadi untuk produk

pembelajaran mesin tidak kedaluwarsa. Jika perubahan harga diperlukan di masa depan, pembeli harus beralih ke penawaran pribadi baru. Instans atau titik akhir produk yang sedang berjalan secara otomatis ditagih dengan tarif per jam yang ditetapkan dalam penawaran baru yang diterima. Pastikan Anda mengaturnya ke tarif per jam untuk produk Anda setelah komponen kontrak apa pun dalam penawaran pribadi berakhir. Menetapkan tarif per jam ini menjadi \$0 memungkinkan pembeli untuk menggunakan produk tanpa biaya perangkat lunak Anda tanpa batas waktu.

- Per inferensi — Penawaran pribadi dapat memiliki tingkat inferensi yang berbeda dari tingkat inferensi yang ditampilkan secara publik, jika Anda telah mengonfigurasi [harga inferensi](#) saat produk Anda digunakan sebagai titik akhir.
- Kontrak — Penawaran pribadi dapat berupa kontrak dengan biaya dimuka tetap untuk jumlah hari tertentu. Pembeli diizinkan untuk menggunakan jumlah instans yang tidak terbatas selama seluruh durasi kontrak. Pada akhir kontrak, setiap contoh yang terus berjalan ditagih dengan tarif per jam yang Anda tetapkan dalam penawaran pribadi. Misalnya, Anda dapat membuat kontrak dengan biaya dimuka tetap selama 365 hari penggunaan tanpa batas. Anda juga menetapkan tarif per jam untuk penawaran pribadi. Ketika pembeli menerima penawaran pribadi ini, mereka membayar biaya di muka itu. Ketika kontrak berakhir, setiap contoh yang masih berjalan ditagih pada tarif per jam itu. Jika Anda menawarkan uji coba pribadi gratis, pastikan Anda menetapkan tarif per jam yang benar setelah masa uji coba gratis berakhir untuk menghindari lisensi abadi gratis.

Anda dapat membuat dan memperluas beberapa penawaran pribadi ke satu pembeli. Pembeli yang Anda memperpanjang penawaran pribadi untuk memiliki opsi untuk memilih antara penawaran pribadi dan penawaran umum. Pembeli hanya dapat berlangganan satu penawaran pada waktu tertentu. Mereka tidak dapat berlangganan penawaran pribadi dan penawaran umum pada saat yang bersamaan.

Untuk membuat penawaran pribadi untuk pembeli tertentu untuk SageMaker produk, lihat [the section called “Membuat penawaran pribadi”](#).

Membuat dan mengelola penawaran pribadi

Sebagai AWS Marketplace penjual, Anda dapat membuat dan mengelola penawaran pribadi. Penawaran pribadi adalah istilah yang dinegosiasikan yang digunakan untuk membeli produk dari AWS Marketplace. Ini dapat melibatkan rencana harga khusus, perjanjian lisensi pengguna akhir (EULA), atau solusi khusus. Bagian berikut menjelaskan cara membuat dan mengelola penawaran pribadi.

Note

Agar memenuhi syarat untuk menerbitkan Penawaran Pribadi, Anda harus memiliki setidaknya satu daftar publik yang aktif. Jika Anda memiliki daftar publik, dan Anda tidak memiliki akses ke tab Penawaran Pribadi, lihat [Izin IAM](#) atau [hubungi AWS Marketplace dukungan](#).

Topik

- [Memulai penawaran pribadi baru](#)
- [Memahami status penawaran](#)
- [Menyusun dan menerbitkan penawaran pribadi](#)
- [Menambahkan penawaran pribadi dan tombol permintaan demo](#)
- [Mengirim penawaran pribadi ke pembeli](#)
- [Mengkloning penawaran pribadi Anda](#)
- [Mengunduh detail penawaran](#)
- [Menyimpan kemajuan penawaran pribadi Anda](#)
- [Memperbarui berakhirnya penawaran pribadi](#)
- [Membatalkan penawaran pribadi](#)

Memulai penawaran pribadi baru

Gunakan proses berikut untuk membuat penawaran dan menghasilkan ID pengguna menggunakan permintaan perubahan API CreateOffer. Ini menciptakan penawaran kosong dalam keadaan draf.

Untuk mulai membuat penawaran pribadi baru

1. Masuk ke [Portal Manajemen AWS Marketplace](#), dan pilih Penawaran.
2. Pada halaman Penawaran, pilih Buat penawaran.
3. Pada halaman Buat penawaran, pilih jenis produk dan produk yang Anda inginkan untuk membuat penawaran pribadi Anda. Pemrosesan membutuhkan waktu hingga 30 detik. Jangan menutup atau menyegarkan halaman sampai pemrosesan selesai.

Note

Anda tidak akan dapat mengubah jenis produk dan produk setelah Anda membuat penawaran. Untuk informasi selengkapnya tentang penawaran pribadi per jenis produk, lihat [Jenis produk yang didukung](#).

Jika Anda adalah Mitra AWS Marketplace Saluran, pilih apakah Anda membuat penawaran untuk produk Anda sendiri atau penawaran pribadi Mitra AWS Marketplace Saluran (CPPO) dari otorisasi penjualan kembali. Jika itu CPPO, pilih vendor perangkat lunak independen (ISV), produk, dan otorisasi.

4. Pilih Lanjutkan untuk menawarkan detail. Sebuah step-by-step pengalaman akan terbuka sehingga Anda dapat terus membuat penawaran pribadi Anda.

Memahami status penawaran

Penawaran memiliki salah satu dari tiga status tergantung pada siklus hidup:

- **Draf** — Penawaran tidak lengkap dan masih dipersiapkan oleh Anda. Penawaran pribadi dalam status draf tidak tunduk pada jadwal retensi. Semua detail yang diperlukan harus diselesaikan dan diserahkan untuk mempublikasikan penawaran dan memperluasnya ke pembeli Anda.
- **Aktif** — Penawaran dipublikasikan dan diperluas ke pembeli. Penawaran belum kedaluwarsa, sehingga pembeli dapat berlangganan penawaran.
- **Kedaluwarsa** — Penawaran dipublikasikan dan diperluas ke pembeli. Penawaran telah kedaluwarsa, sehingga pembeli tidak dapat berlangganan penawaran. Tanggal kedaluwarsa dapat diperbarui untuk memberi pembeli Anda lebih banyak waktu untuk menerima penawaran. Untuk memperbarui kedaluwarsa penawaran, lihat [Memperbarui kedaluwarsa penawaran pribadi](#).

Note

Setelah penawaran diterima, itu akan muncul sebagai perjanjian di tab Perjanjian. Status penawaran tidak akan berubah.

Menyusun dan menerbitkan penawaran pribadi

Gunakan proses berikut untuk menyusun dan mempublikasikan penawaran pribadi Anda.

Untuk menyusun dan mempublikasikan penawaran pribadi Anda

1. Pada halaman Menyediakan informasi penawaran, berikan nama penawaran, detail penawaran, jenis perpanjangan, dan tanggal kedaluwarsa penawaran. Jika ini adalah penawaran perpanjangan, Anda harus memilih salah satu Pelanggan yang Ada untuk perpanjangan yang dimaksudkan AWS Marketplace untuk memperbarui perjanjian yang ada yang dibuat di AWS Marketplace, atau Pelanggan yang Ada Pindah ke AWS Marketplace untuk perpanjangan yang dimaksudkan untuk memigrasikan pelanggan Anda yang sudah ada ke AWS Marketplace

Note

Tanggal kedaluwarsa penawaran adalah tanggal penawaran menjadi batal demi hukum. Setelah 23:59:59 UTC pada tanggal ini, pembeli tidak akan dapat melihat dan menerima penawaran pribadi ini.

2. Pilih Berikutnya.
3. Pada halaman Konfigurasi harga dan durasi penawaran, pilih model harga, durasi kontrak atau penggunaan, harga, mata uang, dan jadwal pembayaran. Untuk model harga yang memiliki paket cicilan, lihat Paket [cicilan](#).

Note

Penawaran pribadi non-USD terbatas pada produk penetapan harga kontrak. Selain itu, pastikan Anda telah mengonfigurasi preferensi pencairan non-USD Anda. Untuk informasi selengkapnya, lihat [the section called “Preferensi pencairan”](#).

Semua penawaran publik dan penawaran pribadi dengan harga konsumsi hanya dapat dibuat dalam USD.

4. Pada halaman Tambah pembeli, berikan Akun AWS ID untuk setiap AWS Marketplace pembeli yang Anda perluas penawaran pribadi. Setiap pembeli yang dipilih harus memiliki Akun AWS di Wilayah AWS mana mata uang penawaran yang dipilih didukung. Untuk menambahkan Akun AWS ID lain, pilih Tambahkan pembeli lain. Anda dapat menambahkan hingga 24 pembeli untuk setiap penawaran pribadi.
5. Pilih Berikutnya.
6. Pada halaman Konfigurasi persyaratan hukum dan penawaran dokumen, pilih salah satu opsi berikut:

- Perjanjian lisensi pengguna akhir penawaran umum (EULA) — Gunakan EULA dari penawaran publik Anda.
 - Kontrak standar untuk AWS Marketplace (SCMP) — Gunakan kontrak standar yang disediakan oleh AWS Marketplace.
 - Ketentuan hukum khusus - Unggah hingga lima file yang terkait dengan penawaran pribadi Anda, termasuk persyaratan hukum, pernyataan kerja, tagihan materi, lembar harga, atau addendum lainnya. File-file ini akan digabungkan menjadi satu dokumen saat penawaran dibuat.
7. Pada halaman Tinjau dan buat, tinjau detail penawaran pribadi Anda. Setelah Anda meninjau dan mengonfirmasi, pilih Buat penawaran untuk mempublikasikan penawaran dan memperluasnya ke pembeli yang Anda pilih. Penerbitan penawaran mencakup permintaan ke API AWS Marketplace Katalog, sehingga dapat memakan waktu hingga satu jam untuk memvalidasi dan memproses penawaran. Permintaan ini dapat dilihat di halaman Permintaan.

 Note

Penawaran akan dipublikasikan dan diperpanjang hanya jika permintaan berhasil. Jika permintaan gagal, itu tidak akan diperluas ke pelanggan. Kegagalan berarti ada kesalahan sistem atau kesalahan yang harus Anda perbaiki sebelum mengirimkan ulang.

Menambahkan penawaran pribadi dan tombol permintaan demo

Penjual dapat menambahkan call-to-action tombol ke halaman detail produk mereka. Tombol-tombol tersebut memungkinkan pembeli untuk meminta penawaran pribadi dan demo produk yang dipandu. Anda dapat menambahkan satu atau kedua tombol ke halaman detail produk Anda.

Anda dapat menggunakan tombol dengan jenis produk berikut:

- Gambar Mesin Amazon
- Perangkat lunak sebagai layanan (SaaS)
- Kontainer
- CloudFormation template

Untuk menggunakan tombol, Anda harus menjadi anggota APN Customer Engagements Program (ACE). Ketika pembeli meminta penawaran atau demo, mereka memasukkan data kontak mereka

dan meminta detail ke dalam formulir. Tim Generasi AWS Permintaan kemudian memenuhi syarat permintaan dan mentransfer permintaan yang memenuhi syarat tersebut kepada Anda sebagai peluang AWS awal melalui ACE di Partner Central. Anda kemudian menindaklanjuti dengan pelanggan untuk mendiskusikan detail penawaran atau menjadwalkan demo yang dipandu. Untuk informasi lebih lanjut tentang ACE, lihat situs web [Program Keterlibatan Pelanggan APN dan Prospek dan Peluang di APN Customer Engagements](#) (ACE). FAQs

Langkah-langkah dalam topik berikut menjelaskan cara menambahkan tombol ke halaman detail produk Anda.

Topik

- [Prasyarat tombol](#)
- [Mengaktifkan tombol](#)

Prasyarat tombol

Sebelum Anda dapat menambahkan call-to-action tombol ke halaman detail produk Anda, Anda harus memiliki prasyarat berikut:

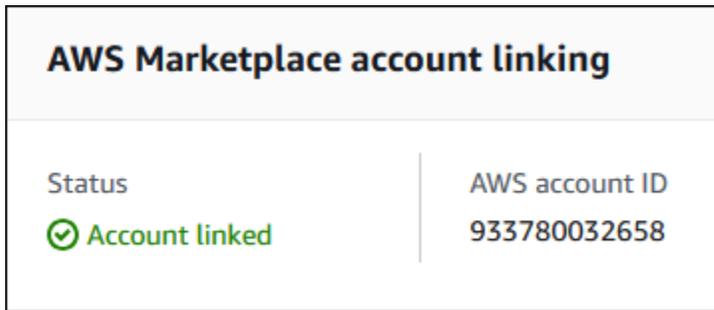
- Pastikan Anda dapat menerima prospek dan peluang yang AWS dirujuk di AWS Partner Central. Untuk informasi selengkapnya, lihat situs web [Program Keterlibatan Pelanggan APN](#).

Note

Setelah Anda mendaftar di program ACE, pembaruan status terjadi setiap dua minggu. Penawaran pribadi dan tombol permintaan demo hanya muncul setelah pembaruan status selesai. Jika Anda memilih opsi ini dan menerima pesan, akses Anda menunggu pembaruan dua minggu berikutnya.

- Tautkan Pusat AWS Mitra Anda dan AWS Marketplace akun. Untuk melakukan itu, Anda harus:
 - Buat kebijakan `CreatePartnerCentralCloudAdminRole` IAM:. Untuk informasi selengkapnya, lihat [prasyarat untuk menautkan akun di Panduan Memulai](#) Pusat AWS Mitra.
 - Tautkan Pusat AWS Mitra Anda dan AWS Marketplace akun. Untuk informasi selengkapnya, lihat [Menautkan akun Pusat AWS Mitra Anda ke AWS Marketplace akun Anda](#), di Panduan Memulai Pusat AWS Mitra.

Setelah Anda menautkan Pusat AWS Mitra dan AWS Marketplace akun, halaman Beranda Pusat Mitra Anda akan menampilkan pesan status berikut:



Untuk informasi selengkapnya, masuk ke Partner Central dan lihat yang berikut ini:

- Panduan [penautan akun AWS Partner Central & Marketplace](#)
- Video [AWS Partner Demo yang Menautkan Akun Marketplace](#)

Note

Anda harus masuk untuk menggunakan sumber daya ini.

Mengaktifkan tombol

Setelah Anda menjadi ACE memenuhi syarat untuk menerima AWS arahan, Anda menggunakan Portal AWS Marketplace Manajemen untuk mengaktifkan satu atau kedua call-to-action tombol.

Anda mengikuti proses terpisah untuk mengaktifkan tombol, tergantung pada apakah Anda membuat daftar produk baru atau memperbarui daftar saat ini.

Untuk mengaktifkan tombol untuk produk baru

1. Gunakan [Portal Manajemen AWS Marketplace](#) untuk membuat jenis produk berikut dan menjadikannya publik:
 - AMI
 - SaaS
 - Kontainer
 - Templat Cloud Front

2. Sebagai bagian dari pembuatan produk, di bawah demo terpandu dan permintaan penawaran pribadi, pilih kombinasi Aktifkan permintaan demo terpandu untuk pembeli dan Aktifkan permintaan penawaran pribadi untuk pembeli.

Note

Tombol-tombol hanya muncul di halaman detail produk dalam penawaran pribadi Anda setelah Anda membuat produk publik.

Untuk mengaktifkan tombol untuk produk yang ada

1. Di tab [Portal Manajemen AWS Marketplace](#), pada tab Produk, pilih produk yang ingin Anda ubah.
2. Buka daftar Permintaan perubahan dan pilih Perbarui informasi produk.
3. Pilih kombinasi Aktifkan permintaan demo terpandu untuk pembeli dan Aktifkan permintaan penawaran pribadi untuk pembeli.

Tombol hanya muncul di halaman detail produk setelah Anda menyimpan perubahan.

Mengirim penawaran pribadi ke pembeli

Setelah penawaran pribadi dipublikasikan, pembeli dapat melihatnya dengan menavigasi ke tab Penawaran pribadi yang tersedia di halaman Penawaran pribadi di Portal Manajemen AWS Marketplace. Pada tab Penawaran pribadi yang tersedia, pembeli dapat melihat penawaran yang diperluas oleh Mitra AWS Marketplace Saluran di kolom Penjual rekaman. Vendor perangkat lunak independen (ISV) akan ditampilkan di kolom Publisher. Pembeli dapat menavigasi ke penawaran pribadi dengan memilih ID Penawaran yang sesuai dalam daftar penawaran mereka.

Pembeli dapat melihat penawaran IDs yang telah diterima atau yang telah kedaluwarsa pada tab Penawaran yang Diterima atau kedaluwarsa.

Setelah penawaran pribadi dipublikasikan, Anda dapat mengirim URL kepada pembeli Anda ke halaman pemenuhan untuk penawaran tersebut.

Untuk mengirim penawaran pribadi kepada pembeli Anda

1. Masuk ke [Portal Manajemen AWS Marketplace](#), dan pilih Penawaran.

2. Pilih tombol radio di sebelah penawaran.
3. Pilih Tindakan dan kemudian Salin URL Penawaran.
4. Kirim URL ke pembeli Anda.

Mengkloning penawaran pribadi Anda

Anda dapat mengkloning penawaran pribadi, termasuk penawaran pribadi Mitra AWS Marketplace Saluran. Gunakan kloning untuk membuat penawaran baru menggunakan templat atau untuk memperbarui dan mengganti penawaran yang ada.

Untuk mengkloning penawaran pribadi

1. Masuk ke [Portal Manajemen AWS Marketplace](#), dan pilih Penawaran.
2. Di tabel Penawaran, pilih opsi di sebelah penawaran yang ingin Anda kloning.
3. Pilih Penawaran klon.
4. Pengalaman pembuatan penawaran baru akan terbuka dengan informasi yang telah diisi sebelumnya dari penawaran yang dipilih. Tinjau dan modifikasi detail penawaran sesuai kebutuhan.
5. (Opsional) Jika Anda mengkloning untuk mengganti penawaran yang ada, pilih Batalkan penawaran yang ada. Ketika dipilih, penawaran asli akan secara otomatis kedaluwarsa dan tidak dapat diakses oleh pembeli ketika penawaran baru ini diterbitkan. Ini hanya memengaruhi aksesibilitas penawaran dan tidak memengaruhi langganan yang ada jika pembeli telah menerima penawaran asli.
6. Pilih Penawaran pribadi Clone. Ini akan mempublikasikan penawaran dan memperluasnya ke pembeli yang Anda pilih sebelumnya.

Mengunduh detail penawaran

Gunakan prosedur berikut untuk mengunduh detail penawaran dalam file.pdf.

Untuk mengunduh detail penawaran

1. Masuk ke [Portal Manajemen AWS Marketplace](#), dan pilih Penawaran.
2. Di tabel Penawaran, pilih opsi di sebelah penawaran dan pilih Lihat detail. Atau, Anda dapat memilih tautan untuk penawaran di kolom ID Penawaran.
3. Pada halaman detail penawaran, pilih Unduh PDF.

Menyimpan kemajuan penawaran pribadi Anda

Gunakan proses berikut untuk menyimpan kemajuan Anda dan melanjutkan nanti.

Untuk menyimpan dan melanjutkan pekerjaan Anda

1. Pada setiap langkah selesai, pilih Simpan dan keluar. Di kotak dialog, konfirmasi bahwa Anda menyimpan konten dalam keadaan draf dan tinjau kesalahan validasi apa pun. Jika ada kesalahan validasi atau detail yang hilang, Anda dapat memilih Perbaiki untuk melanjutkan ke langkah dan menyelesaikan masalah. Saat Anda siap, pilih Simpan dan keluar untuk menyimpan perubahan Anda.

Setelah Anda menyimpan dan keluar, permintaan sedang ditinjau saat sedang diproses. Ini bisa memakan waktu beberapa menit atau jam untuk menyelesaikan pemrosesan. Anda tidak dapat melanjutkan langkah-langkah atau memodifikasi permintaan sampai berhasil. Setelah permintaan berhasil, Anda telah menyelesaikan penyimpanan. Jika permintaan gagal, ada kesalahan sistem atau kesalahan yang harus Anda perbaiki sebelum mengirimkan ulang.

2. Untuk melanjutkan mengerjakan penawaran Anda, buka halaman Penawaran, pilih penawaran Anda, lalu pilih Lanjutkan pembuatan penawaran.
3. Setelah selesai, Anda dapat memilih Simpan dan keluar untuk menyimpan kemajuan Anda atau Buat penawaran untuk mempublikasikan dan memperluas penawaran pribadi ke pembeli pilihan Anda.

Memperbarui berakhirnya penawaran pribadi

Gunakan proses berikut untuk memperbarui tanggal kedaluwarsa layanan pribadi.

Untuk memperbarui tanggal kedaluwarsa penawaran pribadi

1. Masuk ke [Portal Manajemen AWS Marketplace](#), dan pilih Penawaran.
2. Pada halaman Penawaran, pilih penawaran yang ingin Anda perbarui.
3. Pilih Edit.
4. Berikan tanggal kedaluwarsa penawaran baru.
5. Pilih Kirim.

Setelah pembaruan selesai, penawaran akan berubah menjadi status Aktif dan pembeli Anda dapat menerima penawaran tersebut.

Membatalkan penawaran pribadi

Gunakan proses berikut untuk membatalkan penawaran pribadi.

1. Masuk ke [Portal Manajemen AWS Marketplace](#), dan pilih Penawaran.
2. Pada halaman Penawaran, pilih penawaran yang ingin Anda perbarui.

Note

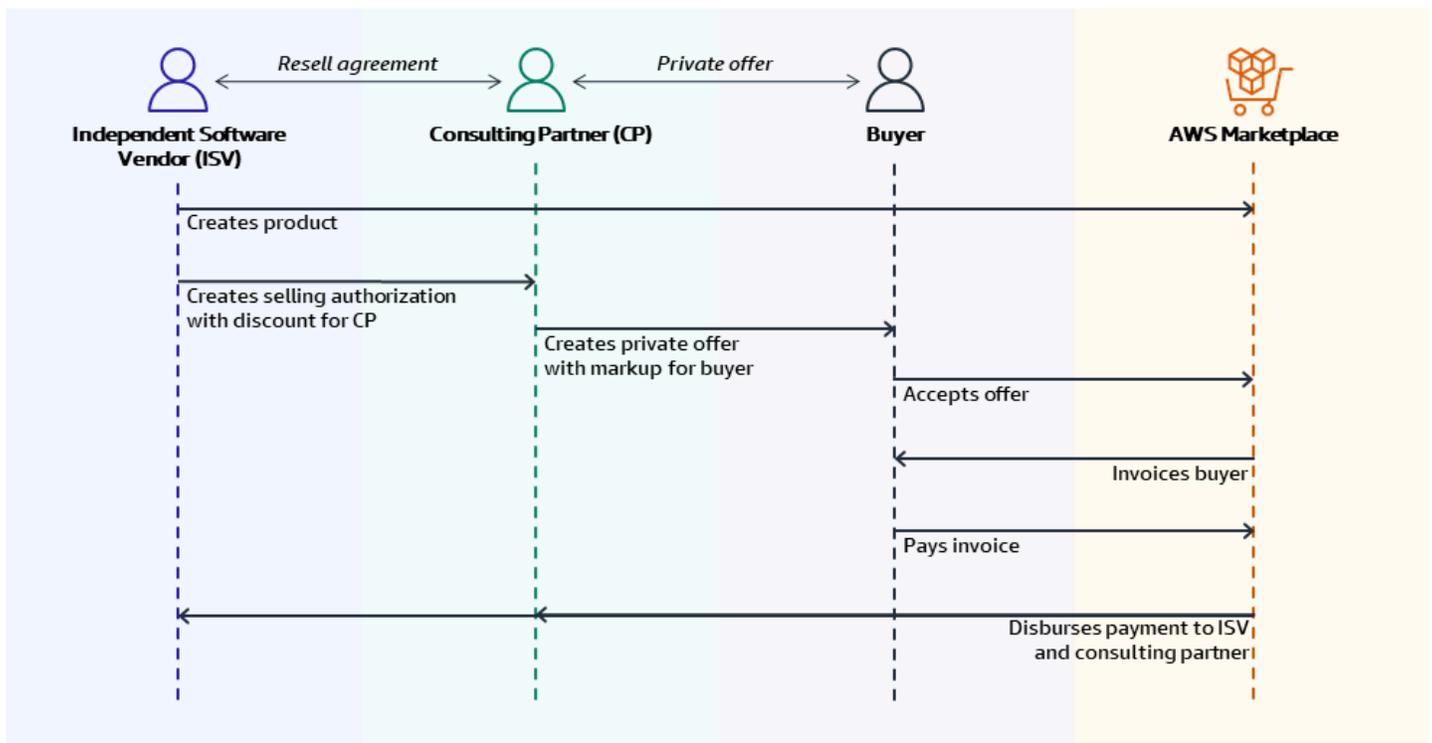
Membatalkan penawaran akan mengubah tanggal kedaluwarsa penawaran, sehingga penawaran akan ditampilkan sebagai kedaluwarsa untuk pembeli yang diperpanjang penawaran ini.

3. Pilih Tindakan dan kemudian pilih Batalkan penawaran.

Membuat penawaran pribadi sebagai Mitra AWS Marketplace Saluran

AWS Marketplace Penawaran pribadi Mitra Saluran memberi mitra saluran kesempatan untuk menjual kembali produk (ISVs) vendor perangkat lunak independen di AWS Marketplace Mitra AWS Marketplace Saluran dan ISV membuat perjanjian untuk menjual kembali satu atau lebih produk ISV, dan kemudian Mitra Saluran memperluas penawaran pribadi kepada pembeli untuk produk tersebut.

Diagram berikut menunjukkan hubungan antara ISV, mitra saluran, dan pembeli.



Note

Untuk informasi selengkapnya tentang membuat otorisasi penjualan untuk mitra saluran, sebagai ISV, lihat. [Membuat otorisasi penjualan untuk Mitra AWS Marketplace Saluran sebagai ISV](#)

Setiap penawaran pribadi Mitra AWS Marketplace Saluran hanya dapat dilihat oleh satu pembeli, dengan harga yang disesuaikan dan persyaratan komersial unik untuk memenuhi kebutuhan pembeli tersebut. Saat membuat penawaran pribadi, Anda mulai dari biaya grosir yang ditetapkan oleh ISV. Kemudian Anda menandai harga itu untuk menciptakan harga penawaran pembeli.

Note

Saat membuat penawaran pribadi, Mitra Saluran harus menggunakan mata uang yang didefinisikan ISV dalam otorisasi penjualan.

Anda menentukan biaya grosir dengan salah satu cara berikut:

- **Recurring discount** — ISV memberi wewenang kepada AWS Marketplace Channel Partner untuk menjual kembali produk atau produk mereka dengan diskon yang disepakati dari harga daftar mereka dengan otorisasi penjualan berulang. AWS Marketplace Channel Partner dapat memanfaatkan diskon ini untuk terus menjual kembali produk tanpa negosiasi harga lebih lanjut dengan ISV. Diskon ini dapat diatur hingga tanggal yang ditentukan, atau tanpa batas waktu, hingga berakhir oleh ISV atau mitra saluran.
- **Non-recurring discount** — Otorisasi penjualan yang diberikan ISV kepada AWS Marketplace Channel Partner adalah diskon satu kali yang dimaksudkan untuk digunakan hanya dengan pembeli tertentu.

Dalam kedua kasus, setelah pembeli membayar untuk penawaran pribadi, AWS Marketplace menggunakan proses standar untuk mendistribusikan dana ke Mitra AWS Marketplace Saluran dan ISV berdasarkan harga yang disepakati.

Tip

ISVs dan Channel Partners dapat menggunakan menu Partner pada [Portal Manajemen AWS Marketplace](#) untuk melihat otorisasi penjualan.

Untuk petunjuk mendetail tentang membuat penawaran pribadi, lihat Penawaran [Pribadi Mitra AWS Marketplace Saluran — Buat Penawaran](#).

Untuk informasi tentang pembiayaan pihak ketiga untuk penawaran pribadi, lihat [Pembiayaan pelanggan sekarang tersedia di AWS Marketplace](#).

Informasi tambahan

Untuk informasi dan pertanyaan tambahan, kami mendorong ISVs dan menyalurkan mitra untuk terhubung dengan tim AWS Marketplace saluran. Jika Anda tidak tahu siapa yang harus dihubungi secara spesifik, kirim pesan email ke aws-mp-channel@amazon.com, dan seseorang di tim akan merespons Anda dalam satu hari kerja.

Membuat otorisasi penjualan untuk Mitra AWS Marketplace Saluran sebagai ISV

Sebagai vendor perangkat lunak independen (ISV), Anda dapat memberi wewenang kepada Mitra AWS Marketplace Saluran untuk menjual kembali produk Anda dengan membuat otorisasi penjualan

untuk mitra tersebut. Mitra Saluran dapat menggunakan otorisasi penjualan untuk membuat Penawaran Pribadi Mitra Saluran (CPPO) kepada pembeli akhir. Jenis produk yang didukung meliputi:

- Produk berbasis AMI
- Produk berbasis kontainer
- Produk berbasis SaaS
- Produk layanan profesional

Prosedur berikut menguraikan cara ISVs membuat otorisasi penjualan untuk Mitra AWS Marketplace Saluran. Untuk menggunakan fitur ini, Anda harus memiliki izin untuk menggunakan tab Otorisasi penjualan di Portal Manajemen AWS Marketplace Untuk informasi selengkapnya, lihat [Kebijakan untuk AWS Marketplace penjual](#).

Buat otorisasi penjualan

1. Masuk ke akun [Portal Manajemen AWS Marketplace](#) dengan AWS Marketplace Penjual Anda.

Tip

Pastikan Anda keluar dari yang lain Akun AWS sebelum masuk dengan akun AWS Marketplace Penjual Anda.

2. Pilih tab Otorisasi penjualan, lalu pilih Buat otorisasi penjualan.
3. Pada halaman Buat otorisasi penjualan, masukkan detail berikut:
 - Untuk Menjual nama otorisasi, masukkan nama untuk otorisasi.

Note

Informasi yang Anda masukkan dalam nama otorisasi penjualan akan terlihat oleh mitra saluran dalam laporan penjual mereka.

- Untuk Reseller, pilih AWS Marketplace Channel Partner (reseller) yang ingin Anda otorisasi dari daftar dropdown. Anda dapat memilih reseller berdasarkan nama atau ID akun.
- Untuk jenis Produk, pilih jenis produk, lalu pilih salah satu Produk Anda yang ingin Anda buat otorisasi penjualan.

- Pilih Lanjutkan ke detail otorisasi.
4. Pada halaman Tentukan detail, masukkan detail berikut:
- Untuk deskripsi otorisasi penjualan, masukkan deskripsi.

 Note

Informasi yang Anda masukkan dalam deskripsi otorisasi penjualan akan terlihat oleh mitra saluran dalam laporan penjual mereka.

- Untuk Pembaruan, tunjukkan apakah otorisasi ini dimaksudkan untuk memperbarui langganan berbayar yang ada dengan pelanggan yang sudah ada untuk produk yang sama.
 - (Opsional) Tetapkan satu atau lebih akun Pembeli IDs untuk menentukan bahwa otorisasi penjualan hanya untuk pembeli tersebut.
 - Anda dapat memilih Simpan dan keluar atau pilih Berikutnya. Memilih Simpan dan keluar pada setiap langkah dalam proses akan menyimpan otorisasi penjualan Anda sebagai draf. Memilih Berikutnya membawa Anda ke halaman Pilih durasi dan harga.
5. Pada halaman Pilih durasi dan harga, masukkan detail berikut:
- Untuk model Harga, pilih salah satu opsi berikut:
 - Harga kontrak dengan paket cicilan - Anda menentukan jumlah tetap untuk setiap jenis dimensi, tarif per jam untuk kelebihan, dan paket cicilan. Kelebihan dibebankan pada tarif per jam dan ditagih secara terpisah.
 - Harga kontrak dengan pembayaran di muka - Anda menentukan harga kontrak untuk setiap jenis dimensi dan tarif per jam untuk penggunaan tambahan. Pembeli memiliki opsi untuk memilih jumlah yang akan dibayarkan pada harga kontrak untuk setiap jenis dimensi dan mendapatkan faktur untuk jumlah penuh setelah diterima. Penggunaan tambahan dibebankan dengan tarif per jam dan ditagih secara terpisah.
 - Harga penggunaan - Anda menentukan tarif per jam untuk setiap jenis dimensi. Pembeli dikenakan biaya berdasarkan tarif per jam untuk penggunaannya.
 - Untuk mata uang, pilih mata uang untuk otorisasi penjualan.

 Note

Mata uang non-USD hanya tersedia untuk penawaran harga kontrak.

- Untuk Durasi, pilih durasi otorisasi penjualan.

 Note

Tanggal mulai pengecer harus lebih awal dari tanggal yang dicantumkan pabrikan dalam otorisasi penjualan kembali.

- Bagian Dimensi Produk menunjukkan dimensi kontrak yang tersedia saat ini dalam produk. Anda dapat memilih dimensi yang ingin Anda sertakan dalam otorisasi penjualan dan memberikan kuantitas atau harga. Harga kontrak dengan paket cicilan memungkinkan Anda memasukkan kuantitas per dimensi, Harga kontrak dengan pembayaran di muka memungkinkan Anda memasukkan harga per dimensi.
- Pilih Kelola dimensi khusus untuk menambahkan dimensi ke produk yang akan tersedia di produk ini untuk penawaran masa depan dan otorisasi penjualan. Dimensi khusus yang ditambahkan ke produk tidak dapat dihapus. Anda dapat memiliki hingga 100 dimensi dalam 1 produk.
- Untuk Harga per dimensi penggunaan, tentukan tarif per jam untuk setiap jenis dimensi. Pembeli dikenakan biaya berdasarkan tarif per jam untuk penggunaannya. Harga penawaran umum diisi di sini secara default.
- Untuk paket cicilan Pembeli, masukkan total Kontrak dan buat rencana angsuran berdasarkan frekuensi yang diinginkan. Ini diperlukan untuk harga Kontrak dengan rencana cicilan.
- Untuk Harga per jenis instans, tetapkan harga penggunaan untuk setiap jenis instans. Alat penetapan harga opsional memungkinkan Anda memperbarui harga secara massal dengan menerapkan diskon ke harga publik atau menerapkan harga yang sama untuk semua instans.
- Untuk ketersediaan otorisasi penjualan, pilih salah satu opsi berikut untuk membatasi ketersediaan berapa banyak penawaran pribadi yang dibuat atau sampai waktu tertentu penawaran pribadi dapat dibuat menggunakan otorisasi penjualan ini:
 - Penggunaan Tunggal - Memungkinkan satu penawaran dibuat oleh pengecer hingga tanggal akhir yang ditentukan
 - Durasi Waktu Tertentu - Memungkinkan beberapa penawaran dibuat oleh pengecer hingga tanggal akhir yang ditentukan
 - Tidak Ada Durasi Waktu yang Ditetapkan - Memungkinkan beberapa penawaran dibuat oleh pengecer hingga otorisasi penjualan dinonaktifkan secara manual
- Anda dapat memilih Simpan dan keluar, Sebelumnya, atau Berikutnya. Memilih Berikutnya akan membawa Anda ke halaman Konfigurasi persyaratan hukum.

6. Pada halaman Konfigurasi persyaratan hukum, masukkan detail berikut:
 - Untuk perjanjian lisensi pengguna akhir, pilih EULA Publik atau Kontrak Standar untuk AWS Marketplace (SCMP) atau unggah EULA Kustom.
-  **Note**
Hanya kustom EULAs yang didukung untuk penjual layanan profesional.
- (Opsional) Pilih Kontrak Reseller untuk AWS Marketplace (RCMP) atau unggah kontrak khusus untuk dimasukkan dalam otorisasi penjualan.
 - Anda dapat memilih Simpan dan keluar, Sebelumnya, atau Berikutnya. Memilih Berikutnya membawa Anda ke halaman Review dan membuat.
 7. Pada halaman Review dan create, pastikan semua informasi sudah benar. Setelah otorisasi penjualan diterbitkan, itu tidak dapat dimodifikasi.
 8. Pilih Buat otorisasi penjualan untuk mempublikasikan otorisasi penjualan ke pengecer.

Kelola otorisasi penjualan

- Tabel yang dibuat otorisasi Penjualan diperbarui untuk menampilkan detail otorisasi penjualan yang relevan termasuk Nama otorisasi penjualan, Nama Produk, Nama Pengecer, Tanggal dibuat, Tanggal kedaluwarsa, dan Status.
- Setelah otorisasi penjualan dibuat, Anda tidak dapat memperpanjang tanggal kedaluwarsa atau mengubah detail lainnya.
- Anda dapat menonaktifkan otorisasi penjualan jika Anda tidak lagi ingin reseller menggunakannya. Ketika Anda menonaktifkan otorisasi penjualan, penawaran baru tidak dapat dibuat menggunakan otorisasi penjualan tersebut. Setiap penawaran yang sudah dibuat tidak terpengaruh.
- Anda juga dapat mengkloning otorisasi penjualan dengan memilih otorisasi penjualan dan memilih Klon. Ini mengisi nilai ke semua bidang dan memungkinkan untuk mengedit.

Status dan tindakan otorisasi penjualan

Daftar berikut menjelaskan nilai status otorisasi penjualan dan artinya:

- Draf - Otorisasi penjualan telah dibuat tetapi belum diaktifkan.

- Resmi - Otorisasi penjualan aktif dan dapat digunakan untuk membuat Penawaran Pribadi Mitra Saluran (CPPOs). CPPOs Belum dibuat.
- Otorisasi (dapat digunakan kembali) - Otorisasi penjualan telah digunakan untuk membuat setidaknya satu CPPO dan dapat digunakan untuk membuat tambahan. CPPOs Status ini umum untuk otorisasi dengan durasi waktu tertentu atau tidak ada durasi waktu yang ditetapkan.
- Otorisasi (dikonsumsi) - Otorisasi penjualan telah sepenuhnya digunakan dan tidak dapat digunakan untuk membuat tambahan. CPPOs Ini biasanya terjadi dengan otorisasi sekali pakai setelah pengecer membuat penawaran pribadi.
- Kedaluwarsa - Tanggal akhir ketersediaan otorisasi penjualan telah berlalu. Itu tidak bisa lagi digunakan untuk membuat CPPOs.
- Dinonaktifkan - Vendor perangkat lunak independen (ISV) telah menonaktifkan otorisasi secara manual. Itu tidak bisa lagi digunakan untuk membuat CPPOs. Status ini disebut "Terbatas" di API.

Note

Sistem menentukan status berdasarkan beberapa faktor, termasuk status internal otorisasi penjualan, tanggal akhir ketersediaan, dan status perpanjangan penawaran. Nilai status ditetapkan berdasarkan urutan prioritas untuk menyelesaikan potensi tumpang tindih.

Paket cicilan penawaran pribadi

Sebagai AWS Marketplace penjual, Anda dapat menggunakan paket cicilan (juga dikenal sebagai jadwal pembayaran fleksibel) untuk memperpanjang penawaran pribadi dengan jadwal pembayaran khusus. Paket cicilan tersedia untuk penawaran pribadi pada produk dan jenis harga tertentu. Untuk informasi selengkapnya, lihat [Jenis produk yang memenuhi syarat untuk penawaran pribadi](#). Jadwal pembayaran dapat tersebar selama durasi kontrak yang diterima, dengan pembeli melakukan pembayaran dengan angsuran reguler.

Untuk produk Amazon Machine Image (AMI) multi-tahun dan berdurasi khusus, tetapkan jumlah instans untuk setiap jenis instans yang disertakan dalam penawaran dan harga per jam untuk instans tambahan yang diluncurkan. Setelah pembeli meluncurkan jumlah instans yang ditentukan, setiap instans tambahan yang diluncurkan dibebankan pada tarif per jam yang ditentukan dalam penawaran pribadi.

Anda tidak dapat mengubah jadwal pembayaran pada penawaran pribadi yang telah diperpanjang dan berlangganan oleh pembeli. Untuk membuat perubahan, Anda harus [membuat penawaran baru](#).

Membuat paket angsuran untuk penawaran pribadi

Saat membuat penawaran pribadi, Anda dapat mengatur jadwal pembayaran khusus dengan paket angsuran.

Untuk membuat rencana angsuran untuk penawaran pribadi

1. Pada halaman Konfigurasi harga dan durasi penawaran, untuk harga Produk, pilih Harga kontrak dengan paket cicilan.
2. Pilih durasi kontrak untuk penawaran ini dan tentukan detail penawaran. Untuk informasi selengkapnya, lihat [Mempersiapkan penawaran pribadi untuk AWS Marketplace produk Anda](#).
3. Di bawah paket cicilan Pembeli, masukkan parameter berikut:

- Total kontrak
- (Opsional) Jika Anda ingin pembayaran pertama berbeda dari yang lain, masukkan Pembayaran awal. Saldo yang tersisa akan dibagi rata di antara pembayaran berikutnya.
- Frekuensi

Pilih Bulanan, Triwulanan, Tahunan, atau Kustom. Jika Anda memilih Custom, masukkan juga Jumlah angsuran.

Anda dapat menambahkan hingga 60 pembayaran. Anda juga memiliki opsi untuk melakukan penyesuaian pada setiap item jalur pembayaran. Setiap kali Anda menyesuaikan item jalur pembayaran, Jumlah total yang jatuh tempo dari pembeli diperbarui.

- Tanggal faktur pertama
4. Pilih Hasilkan paket cicilan. Anda akan menerima pesan kesalahan jika tanggal faktur berada di luar durasi kontrak.
 5. Setelah Anda memverifikasi semua jumlah dan tanggal faktur, konfirmasi bahwa Jumlah total yang jatuh tempo dari pembeli sesuai dengan harga total yang Anda ingin pembeli Anda bayar selama penawaran pribadi. Untuk menyelesaikan pembuatan penawaran pribadi, selesaikan langkah-langkah yang tersisa [the section called “Menyusun dan menerbitkan penawaran pribadi”](#).

Setelah pembeli menerima penawaran pribadi, mereka akan ditagih pada pukul 00:00 UTC pada tanggal faktur yang Anda tentukan dalam jadwal pembayaran. Anda menerima pembayaran untuk

setiap faktur setelah AWS Marketplace menerima pembayaran dari pembeli. Hanya satu tanggal faktur yang dapat terjadi sebelum tanggal penerimaan penawaran. Jika penawaran pribadi diterima setelah tanggal faktur pertama dalam jadwal pembayaran, faktur pertama akan dibuat segera setelah penawaran diterima. Setelah pembeli Anda berlangganan, mereka dapat melihat semua pembayaran sesuai jadwal dan AWS faktur mereka, membantu mereka melacak pengeluaran mereka.

Pelaporan rencana angsuran

Pelaporan untuk penawaran pribadi dengan paket angsuran muncul di bagian 4 dari laporan pendapatan tagihan bulanan. Untuk informasi lebih lanjut, lihat [Bagian 4: Kontrak dengan jadwal pembayaran yang fleksibel](#).

Membuat perjanjian bertanggal future untuk penawaran pribadi

Sebagai AWS Marketplace penjual, Anda dapat menggunakan perjanjian bertanggal masa depan untuk menjual produk yang akan diterima pembeli pada tanggal masa depan yang telah ditentukan. Dalam AWS Marketplace transaksi tipikal, pembeli menerima lisensi atau hak produk segera setelah penawaran diterima atau perjanjian dibuat. Sebaliknya, dengan future dated agreement (FDA), pembeli menerima lisensi produk atau hak pada tanggal future yang telah ditentukan sebelumnya. FDA dapat digunakan untuk mengatur pembaruan untuk transaksi yang ada dengan pembeli. FDA didukung untuk produk perangkat lunak sebagai layanan (SaaS) untuk kontrak dan kontrak dengan harga konsumsi (CCP), dengan dan tanpa pembayaran yang fleksibel. Bagian berikut memberikan informasi tentang bekerja dengan perjanjian bertanggal future.

Dengan FDA, Anda dapat menutup transaksi dengan pembeli ketika mereka memilih daripada ketika pembeli ingin memulai penggunaan produk. Anda dapat menggunakan FDA untuk secara dependen melakukan tindakan berikut untuk transaksi pada: AWS Marketplace

- Pesan (pembeli menerima penawaran) kesepakatan berdasarkan kebutuhan penjualan.
- Membebaskan biaya pembeli berdasarkan kebutuhan keuangan atau akuntansi Anda.
- Memberikan akses pembeli ke produk, seperti mengaktifkan lisensi atau hak, berdasarkan kebutuhan pembeli.

Pembeli dapat menerima penawaran FDA, bahkan saat perjanjian mereka saat ini aktif, jika:

- Tanggal mulai FDA terjadi setelah perjanjian yang ada berakhir.
- Perpanjangan otomatis dinonaktifkan sebelum menerima FDA.

⚠ Important

Setelah FDA diterima, perpanjangan otomatis tidak dapat diaktifkan kembali.

- Tanggal layanan tidak tumpang tindih dengan yang diterima FDAs lainnya.

Topik

- [Pertimbangan untuk perjanjian data future](#)
- [Membuat perjanjian bertanggal masa depan](#)
- [Menggunakan rencana angsuran dengan perjanjian bertanggal masa depan](#)
- [Menerima pemberitahuan untuk perjanjian bertanggal masa depan](#)
- [Menggunakan perjanjian bertanggal future dengan penjualan kembali untuk penawaran pribadi Channel Partner](#)

Pertimbangan untuk perjanjian data future

Saat Anda menggunakan perjanjian bertanggal masa depan, ingatlah tanggal-tanggal berikut:

Tanggal penandatanganan perjanjian

Tanggal ketika pembeli menerima penawaran dan kapan perjanjian dibuat.

Tanggal mulai perjanjian

Tanggal ketika lisensi pembeli atau hak atas produk diaktifkan dan pembeli dapat mulai menggunakan produk.

Tanggal akhir perjanjian

Tanggal ketika perjanjian berakhir. Perjanjian dan lisensi atau hak pembeli kedaluwarsa pada tanggal ini.

📘 Note

Istilah kontrak yang ditentukan dalam perjanjian lisensi pengguna akhir Anda, formulir pemesanan, atau kontrak lain dengan pelanggan akan mengontrol jika ada konflik dengan istilah yang ditentukan dalam faktur pelanggan AWS Marketplace .

Membuat perjanjian bertanggal masa depan

Penjual catatan menetapkan tanggal mulai perjanjian saat menghasilkan penawaran pribadi dengan tanggal mulai masa depan. Pembeli tidak dapat mengubah tanggal mulai, tetapi mereka dapat meninjau tanggal mulai sebelum menerima penawaran pribadi di AWS Marketplace.

Untuk membuat penawaran pribadi dengan tanggal mulai future

1. Saat membuat penawaran pribadi, pilih Mulai di masa depan di bawah Durasi kontrak.
2. Di bagian Tanggal Layanan, masukkan tanggal mulai Layanan dan tanggal akhir Layanan. Tanggal mulai layanan yang Anda pilih di sini akan menjadi tanggal mulai perjanjian perjanjian masa depan Anda saat pembeli menerima penawaran tersebut.

Note

Untuk menggunakan FDA untuk perpanjangan, tanggal mulai layanan harus satu hari atau lebih setelah tanggal akhir perjanjian yang ingin Anda perbarui. Misalnya, jika tanggal akhir perjanjian adalah 12/31/2024, Anda harus menetapkan tanggal mulai layanan menjadi 1/1/2025.

Penjual dapat memilih tanggal mulai layanan hingga 3 tahun di masa depan.

Menggunakan rencana angsuran dengan perjanjian bertanggal masa depan

Dengan menggunakan paket angsuran dengan FDA, Anda dapat mengatur pembayaran untuk pembelian yang terjadi kapan saja antara tanggal penandatanganan perjanjian dan tanggal akhir perjanjian. Ini termasuk pembayaran sebelum dan sesudah tanggal mulai perjanjian.

Penjual catatan memilih tanggal dan jumlah pembayaran penawaran pribadi. Untuk detail selengkapnya tentang menyiapkan paket cicilan, lihat [the section called “ Membuat paket angsuran untuk penawaran pribadi”](#).

Menerima pemberitahuan untuk perjanjian bertanggal masa depan

Anda menerima [pemberitahuan email](#) ke akun root yang ditunjuk untuk tindakan berikut yang diambil pada perjanjian bertanggal masa depan Anda:

- Penerimaan penawaran/pembuatan perjanjian (tanggal tanda perjanjian)
- Setelah lisensi atau aktivasi hak (tanggal mulai perjanjian)

- Peningkat untuk perjanjian yang kedaluwarsa 30, 60, atau 90 hari sebelumnya
- Kedaluwarsa perjanjian (tanggal akhir perjanjian)
- Setelah amandemen perjanjian atau penggantian

 Note

Semua notifikasi Amazon Simple Notification Service (Amazon SNS) yang ada untuk SaaS juga berfungsi untuk FDA. Untuk FDAs, kedua topik Amazon SNS dimulai pada tanggal mulai perjanjian (dan bukan tanggal tanda perjanjian). Untuk informasi selengkapnya, lihat [the section called “Notifikasi Amazon SNS untuk produk SaaS”](#).

Menggunakan perjanjian bertanggal future dengan penjualan kembali untuk penawaran pribadi Channel Partner

Produsen dan reseller dapat menggunakan perjanjian bertanggal future untuk penawaran pribadi AWS Marketplace Channel Partner.

Sebagai produsen:

- Mirip dengan Penawaran Pribadi Mitra AWS Marketplace Saluran standar (CPPOs), produsen harus memberi wewenang kepada Mitra AWS Marketplace Saluran untuk membuat CPPOs dengan tanggal mulai di masa mendatang dengan memperpanjang otorisasi penjualan kembali kepada mereka.

Untuk mempelajari cara membuat otorisasi penjualan kembali, ikuti langkah-langkah di halaman [the section called “Membuat otorisasi penjualan sebagai ISV”](#)

- Saat membuat otorisasi penjualan kembali, produsen secara opsional dapat memilih untuk menentukan tanggal mulai layanan maksimum yang diizinkan. Ini akan menjadi tanggal mulai layanan maksimum yang dapat ditentukan oleh Mitra AWS Marketplace Saluran saat membuat penawaran pribadi Mitra AWS Marketplace Saluran yang sesuai.

 Note

Jika pabrikan tidak menentukan tanggal maksimum, Mitra AWS Marketplace Saluran dapat menentukan tanggal layanan future hingga 3 tahun di masa mendatang.

Sebagai reseller:

- Untuk reseller dan Channel Partners, langkah-langkah untuk membuat penawaran pribadi Mitra Channel bertanggal masa depan dan penawaran pribadi bertanggal masa depan yang biasa adalah sama, dengan satu perbedaan utama. Tanggal mulai perjanjian yang dapat ditentukan oleh pengecer harus lebih awal dari yang ditentukan sebagai tanggal mulai layanan maksimum yang diizinkan dalam otorisasi penjualan kembali oleh produsen.
- Untuk mempelajari cara membuat penawaran pribadi Mitra Saluran, lihat [the section called “Penawaran pribadi mitra saluran”](#).

FAQ penawaran pribadi

FAQ ini menjawab pertanyaan umum tentang membuat, mengelola, dan memecahkan masalah penawaran pribadi.

Topik

- [Apa itu Tim Sukses Penawaran Pribadi \(POST\), dan bagaimana penjual dapat menghubungi mereka?](#)
- [Bagaimana penjual dapat mengatasi kesalahan saat mengakses tab Penawaran?](#)
- [Apa itu rencana angsuran atau jadwal pembayaran yang fleksibel?](#)
- [Bagaimana saya bisa membantu pembeli yang menerima Kesalahan 404 saat mengakses penawaran pribadi?](#)
- [Mengapa pelanggan melihat kesalahan “Anda sudah memiliki kontrak aktif” ketika mencoba menerima penawaran pribadi?](#)
- [Bisakah penjual atau pembeli membatalkan penawaran pribadi?](#)
- [Bagaimana cara meminta pengembalian dana atau pembatalan kontrak?](#)
- [Kapan pembeli ditagih?](#)
- [Langkah apa yang harus diambil penjual setelah penawaran diterima?](#)
- [Bagaimana cara AWS membayar penjual dan mitra?](#)
- [Bagaimana cara AWS menilai pajak?](#)
- [Sumber daya dan dukungan](#)

Apa itu Tim Sukses Penawaran Pribadi (POST), dan bagaimana penjual dapat menghubungi mereka?

POST memungkinkan pemirsa eksternal pada pengalaman Penawaran AWS Marketplace Pribadi. Penjual dapat kami [Formulir Dukungan](#) di Portal AWS Marketplace Manajemen. Untuk bantuan dengan formulir dukungan, unduh dan lihat PDF [panduan formulir dukungan penawaran pribadi](#).

Bagaimana penjual dapat mengatasi kesalahan saat mengakses tab Penawaran?

Jika Anda mengalami kesalahan saat memilih tab Penawaran di Portal AWS Marketplace Manajemen, pastikan Anda memenuhi prasyarat berikut:

Prasyarat untuk membuat penawaran pribadi untuk menjual perangkat lunak atau layanan secara langsung

1. Pastikan AWS akun Anda memiliki kebijakan Identity and Access Management (IAM) yang sesuai. Untuk informasi selengkapnya tentang kebijakan yang diperlukan, lihat [kebijakan IAM untuk penawaran pribadi](#).

Note

Jika Anda memerlukan bantuan untuk mengubah kebijakan atau izin IAM, hubungi administrator internal Anda. AWS tidak dapat membantu dengan kebijakan atau izin IAM, karena akses dikelola oleh pelanggan sesuai dengan Model [Tanggung Jawab Bersama](#).

2. Memiliki setidaknya satu produk yang terdaftar secara publik bukan nol. Anda dapat memverifikasi ini dengan mencari vendor Anda di AWS Marketplace. Jika tidak ada produk yang muncul, Anda mungkin belum mendaftarkan produk secara publik atau Anda memiliki daftar terbatas.

Prasyarat untuk membuat penawaran pribadi mitra saluran (CPPOs) untuk menjual kembali perangkat lunak atau layanan

- Memenuhi prasyarat geografis untuk menjadi “penjual” di: AWS Marketplace

- Gunakan badan hukum dari [yurisdiksi yang memenuhi syarat](#), atau badan usaha yang tergabung dalam salah satu bidang tersebut.
- Berikan informasi yang diperlukan tentang rekening bank Anda. Anda dapat menyediakan satu atau beberapa rekening bank, termasuk:
 - Akun Rumah Kliring Otomatis AS (ACH)
 - Rekening bank Society for Worldwide Interbank Financial Telecommunication (SWIFT) dari yurisdiksi yang memenuhi syarat
 - Akun [Hyperwallet](#)

Apa itu rencana angsuran atau jadwal pembayaran yang fleksibel?

Paket cicilan, juga dikenal sebagai jadwal pembayaran fleksibel (FPS), memungkinkan Anda untuk memperpanjang penawaran pribadi dengan jadwal pembayaran khusus. Paket ini tersedia untuk penawaran pribadi pada produk dan jenis harga tertentu. Untuk informasi selengkapnya, lihat [Jenis produk yang memenuhi syarat untuk penawaran pribadi](#).

Jadwal pembayaran dapat tersebar selama durasi kontrak yang diterima, dengan pembeli melakukan pembayaran dengan angsuran reguler. Setelah berlangganan, pelanggan Anda dapat melihat semua pembayaran sesuai jadwal dan AWS faktur mereka, membantu mereka melacak pengeluaran mereka.

Paket cicilan dan FPSs izinkan penjual untuk menentukan:

- Jumlah unit per dimensi atau per jenis instance
- Ketentuan pembayaran untuk kontrak (di muka, tertunda, atau beberapa faktur)

Membuat rencana angsuran

Note

Dalam skenario penjualan kembali perangkat lunak, vendor perangkat lunak independen (ISV) menentukan rencana angsuran.

Untuk membuat rencana angsuran

1. Pada halaman Konfigurasi harga dan durasi penawaran, untuk harga Produk, pilih Harga kontrak dengan paket cicilan.
2. Pilih durasi kontrak dan tentukan detail penawaran.
3. Di bawah rencana angsuran Pembeli, masukkan parameter yang diinginkan:
 - Untuk faktur di muka setelah penerimaan, masukkan jumlah dolar dan atur tanggal faktur ke tanggal Anda membuat penawaran pribadi.
 - Untuk faktur tertunda, masukkan jumlah dolar dan tetapkan tanggal faktur future.
 - Untuk faktur cicilan, pilih Tambahkan Pembayaran untuk memasukkan beberapa item jalur pembayaran dengan jumlah dolar dan tanggal faktur.

Untuk informasi selengkapnya, lihat [Membuat Paket Cicilan](#). Anda juga dapat menonton [video tutorial tentang Paket Cicilan](#).

Mengatur SKU atau tipe instans tetap

Di bagian Dimensi pada halaman pembuatan penawaran:

1. Pilih “Pembeli dapat memilih satu atau lebih opsi yang ditawarkan.”
2. Masukkan dimensi yang berbeda bersama dengan kuantitas untuk masing-masing.
3. Pilih “Tambahkan dimensi” untuk menyertakan beberapa dimensi atau “Buat dan tambahkan dimensi baru” untuk membuat dimensi khusus untuk penawaran pribadi ini.

Bagaimana saya bisa membantu pembeli yang menerima Kesalahan 404 saat mengakses penawaran pribadi?

Berikut adalah alasan umum untuk 404 kesalahan dan resolusinya:

Asosiasi akun salah

1. Minta pembeli untuk mengakses tab [Penawaran Pribadi](#) di [AWS Konsol](#) mereka.
2. Jika penawaran tidak terlihat di penawaran pribadi yang tersedia atau tab Penawaran yang Diterima dan kedaluwarsa:
 - Verifikasi pembeli masuk ke akun yang benar. Mereka dapat mengonfirmasi ID akun mereka di sudut kanan atas AWS Konsol.

- Jika masuk ke akun yang benar, pastikan penawaran pribadi untuk daftar terbatas Amazon Machine Image (AMI) (lihat bagian daftar terbatas AMI di bawah).

Resolusi: Pembeli harus masuk ke akun yang benar, atau Anda perlu mengeluarkan penawaran pribadi ke ID AWS akun yang benar.

Penawaran pribadi yang kedaluwarsa

- Jika penawaran muncul di tab Penawaran yang Diterima dan kedaluwarsa, penawaran tersebut telah kedaluwarsa.

Resolusi: Memperpanjang tanggal kedaluwarsa atau mengeluarkan penawaran pribadi baru.

Produk daftar terbatas AMI

- Jika penawaran tidak terlihat di salah satu tab dan pembeli masuk ke akun yang benar, produk mungkin dalam status daftar terbatas.

Resolusi:

- Untuk daftar terbatas AMI Tunggal: Izinkan daftar akun pembeli dengan mengikuti petunjuk untuk akun pembeli daftar [yang diizinkan](#).
- Untuk daftar terbatas jenis server lainnya: Hubungi [Tim Sukses Penawaran Pribadi \(POST\)](#) untuk mengizinkan daftar akun pembeli.

Note

Allow-listing adalah aktivitas satu kali per produk.

Pembatasan pasar pribadi

- Jika pembeli memiliki pasar pribadi, mereka mungkin melihat kesalahan yang menyatakan produk tidak tersedia di katalog pribadi mereka.

Resolusi: Pembeli perlu menambahkan produk ke daftar izin mereka. lihat [Menambahkan produk ke pasar pribadi](#).

Jika masalah berlanjut, mintalah pembeli untuk mencoba yang berikut:

- Keluar dan masuk kembali
- Bersihkan cache browser mereka
- Hapus cookie
- Masuk ke jendela penyamaran
- Gunakan browser yang berbeda (bukan Internet Explorer)

Mengapa pelanggan melihat kesalahan “Anda sudah memiliki kontrak aktif” ketika mencoba menerima penawaran pribadi?

Kesalahan ini terjadi ketika akun pembeli sudah memiliki langganan aktif ke suatu produk. Resolusi tergantung pada jenis produk:

Untuk kontrak Perangkat Lunak sebagai Layanan (SaaS)

Setiap akun pembeli hanya dapat memiliki satu langganan aktif untuk kontrak SaaS atau kontrak dengan produk konsumsi. Untuk memperbarui atau memperluas langganan aktif:

1. Buat penawaran berbasis perjanjian dari tab Perjanjian di [Portal AWS Marketplace Manajemen \(AMMP\)](#).
2. Sertakan ketentuan pembayaran yang tertunda dari penawaran asli dalam penawaran berbasis perjanjian, karena akan mengesampingkan sisa pembayaran yang belum ditagih.

Membuat penawaran berbasis perjanjian

Untuk Penawaran Pribadi Marketplace (MPPO):

1. Arahkan ke tab Perjanjian di [AMMP](#).
2. Cari ID akun pembeli.
3. Pilih perjanjian pembeli saat ini.
4. Pilih “Buat Penawaran Berbasis Perjanjian.”

Untuk Penawaran Pribadi Mitra Saluran (CPPO):

1. ISV memberikan otorisasi penjualan kepada mitra saluran dengan harga baru dari tab Mitra mereka, termasuk jumlah unit lisensi, ketentuan pembayaran ke CP, dan durasi kontrak.
2. Mitra saluran menavigasi ke tab Perjanjian di [AMMP](#).
3. Cari ID akun pembeli.
4. Pilih perjanjian pembeli saat ini.
5. Pilih Buat Penawaran Berbasis Perjanjian.
6. Buka daftar Otorisasi Jual dan pilih peluang baru dari langkah 1. Hal ini memungkinkan mitra saluran untuk memasukkan persyaratan untuk penawaran berbasis perjanjian.

Untuk informasi selengkapnya, lihat [Membuat penawaran berbasis perjanjian](#).

Opsi lain untuk kontrak dan kontrak SaaS dengan harga konsumsi (CCPs):

- Buat penawaran pribadi baru untuk ID produk yang sama yang menargetkan akun pembeli yang berbeda.
- Menerbitkan penawaran pribadi baru untuk ID produk yang berbeda ke akun pembeli yang sama.
- Minta pembatalan kontrak aktif, lalu keluarkan penawaran pribadi baru untuk ID produk dan ID pembeli yang sama.

Untuk AMI setiap jam dan tahunan

Setiap akun pembeli hanya dapat memiliki satu langganan aktif. Untuk memperbarui atau memperluas:

1. Keluarkan penawaran baru dari tab Penawaran.
2. Ketika pembeli menerima, persyaratan baru mengesampingkan ketentuan kontrak sebelumnya.
3. Biaya yang tertunda dari jadwal pembayaran penawaran sebelumnya akan berlanjut kecuali dibatalkan.

Untuk kontrak AMI dan layanan profesional

Langganan pembeli saat ini harus dibatalkan sebelum menerima penawaran pribadi baru. Gunakan [formulir pengembalian uang/pembatalan untuk memulai permintaan pembatalan](#).

Bisakah penjual atau pembeli membatalkan penawaran pribadi?

- Jika pembeli belum berlangganan: Penjual dapat membatalkan dengan menavigasi ke tab Penawaran di Portal AWS Marketplace Manajemen, memilih penawaran, dan memilih Batal.
- Jika pembeli telah berlangganan: Penjual catatan harus memulai permintaan pembatalan dengan menggunakan formulir [pengembalian uang/pembatalan di Portal Manajemen](#). AWS Marketplace Untuk petunjuk rinci, lihat [tutorial video](#) ini.

Bagaimana cara meminta pengembalian dana atau pembatalan kontrak?

Pengembalian uang dan pembatalan kontrak ditangani oleh tim Layanan Pelanggan. AWS

Penjual— ISVs untuk penawaran pribadi Marketplace, mitra saluran untuk penawaran pribadi mitra saluran—harus menggunakan [formulir pengembalian/pembatalan untuk memulai pengembalian dana atau pembatalan](#).

[Video ini](#) menjelaskan seluruh proses.

Penjual harus memasukkan data berikut.

- ID akun Pembeli: Informasi ini dapat ditemukan di detail penawaran atau di Dasbor Pendapatan Tagihan. Ini harus ID akun pelanggan.
- ID akun penjual: Ini adalah ID AWS akun penjual yang digunakan untuk membuat penawaran pribadi.
- ID Produk: Anda dapat menemukan informasi ini di detail penawaran atau [Dasbor pendapatan yang ditagih](#) di kolom Legacy Product ID.
- Tanggal penagihan: Anda dapat menemukan informasi ini di detail penawaran atau [Dasbor pendapatan yang ditagih](#) di kolom Periode Mulai Penggunaan.
- Jumlah pengembalian dana: Jika pengembalian dana tidak diperlukan, penjual dapat mengaturnya menjadi \$0.
- Detail tambahan: Lihat catatan berikut.

Important

- Untuk permintaan yang mencakup pembatalan kontrak, sertakan teks berikut”

Harap batalkan langganan akun X ke Offer-X.

Untuk pengembalian uang, menentukan ID AWS faktur pembeli di bagian ini membantu tetapi tidak wajib.

- Simpan ID referensi yang diberikan setelah penyerahan untuk referensi future jika terjadi tindak lanjut.

Setelah Anda mengirimkan permintaan, periksa [AWS Support Console](#) untuk pembaruan status.

Kapan pembeli ditagih?

- Setelah diterima: Faktur dibuat di konsol penagihan segera setelah berlangganan.
- Jadwal pembayaran yang fleksibel: Faktur didasarkan pada jadwal pembayaran khusus yang dinegosiasikan antara penjual dan pembeli.
- AWS tagihan konsolidasi (2/3 setiap bulan): Dapat mencakup pembelian penawaran umum dan produk dengan pengukuran.

Langkah apa yang harus diambil penjual setelah penawaran diterima?

1. Lacak penerimaan penawaran.
2. Untuk tim keuangan: Praktik standar adalah agar penjual menangani pelacakan pembelian dengan membuat pembayaran terbuka dan menekan pembuatan faktur kepada pelanggan untuk pesanan. AWS Marketplace

Bagaimana cara AWS membayar penjual dan mitra?

- Pencairan dimulai hanya setelah dana berhasil dikumpulkan dari pelanggan.
- Pencairan terjadi pada irama pencairan default adalah bulanan antara tanggal 7 dan 10 setiap bulan. Atau, ISV dan mitra saluran dapat memilih jadwal pencairan — harian atau bulanan. Jika ISV atau mitra saluran memilih opsi bulanan, mereka dapat memilih hari dalam bulan yang ingin mereka terima.
- Pencairan disetorkan ke rekening bank AS di ISV atau akun mitra saluran dikurangi biaya listing. Dana dapat memakan waktu 1-3 hari kerja untuk mendarat dengan bank umum ACH SLAs. Jika Anda tidak memiliki rekening bank AS, Anda dapat menggunakan [Hyperwallet](#) untuk menerima

pencairan penjualan Amazon Anda ke rekening deposito dan mentransfernya langsung ke bank lokal Anda dalam mata uang lokal Anda.

Bagaimana cara AWS menilai pajak?

AWS Marketplace membebankan pajak berdasarkan:

- Alamat [pajak pelanggan produk](#)
- Tipe produk
- Hukum Marketplace Facilitator

Marketplace Facilitator: Mewajibkan operator pasar untuk membebankan, mengumpulkan, dan mengirimkan pajak kepada otoritas perpajakan.

Fasilitator Non-Marketplace: Tanggung jawab jatuh pada penjual.

Untuk informasi lebih lanjut tentang kewajiban pajak, lihat [AWS Marketplace — Bantuan Pajak untuk Penjual](#).

Sumber daya dan dukungan

Jika Anda memiliki permintaan khusus, hubungi salah satu AWS Marketplace tim berikut melalui Portal AWS Marketplace Manajemen.

Tim Sukses Penawaran Pribadi (POST): Tim mendukung penjual (ISV dan Mitra Saluran) dan pembeli dengan pemberdayaan dan dukungan operasional penawaran pribadi. Hubungi kami melalui [Support Form](#).

Vendor Finance Success Team (VFS): Tim meningkatkan proses yang memengaruhi aplikasi kas Vendor Finance, rekonsiliasi dan pelaporan terkait, serta mendukung penjual orientasi (ISV dan Mitra Saluran) dengan pertanyaan spesifik keuangan mengenai dan bukan penawaran umum. MPPOs CPPOs Hubungi kami melalui [Support Form](#).

Operasi Katalog Terkelola (MCO): Tim bertanggung jawab untuk memasukkan penjual perangkat lunak pihak ketiga ke AWS Marketplace platform, meninjau dan memproses produk perangkat lunak mereka untuk kepatuhan kebijakan dan pengalaman pembeli, dan mengelola hubungan operasional dengan penjual. Hubungi kami melalui [Support Form](#).

Jika Anda baru mengenal proses penawaran pribadi, gunakan [pustaka video](#) ini untuk membantu Anda memulai dengan pengenalan, ikhtisar, dan jawaban atas pertanyaan yang paling umum. Anda juga dapat menemukan jawaban atas pertanyaan tentang [faktur pembeli](#), [pencairan](#), atau [aturan AWS pajak di video yang](#) ditautkan atau di PDF Sukses Keuangan [AWS Marketplace Vendor](#).

Menggunakan perjanjian

AWS Marketplace perjanjian penjual adalah kontrak antara penjual dan pembeli untuk produk yang dijual AWS Marketplace. Perjanjian terbentuk ketika pembeli menerima penawaran, yang dapat bersifat publik atau pribadi. Topik di bagian ini menjelaskan jenis perjanjian, status, dan cara mengubah atau mengubah perjanjian.

Topik

- [Jenis perjanjian](#)
- [Bekerja dengan perjanjian](#)
- [Mengubah perjanjian di AWS Marketplace](#)

Jenis perjanjian

AWS Marketplace menyediakan jenis perjanjian berikut:

- Perjanjian penawaran umum — Perjanjian hukum standar yang tidak dinegosiasikan yang diterima pembeli saat membeli produk yang terdaftar secara publik dari penjual. AWS Marketplace
- Perjanjian penawaran pribadi — Penawaran pribadi yang diterima pembeli menjadi perjanjian. Untuk informasi lebih lanjut, lihat [Membuat dan mengelola penawaran pribadi](#).
- Perjanjian bertanggal di masa depan - Pembeli menerima lisensi produk atau hak pada tanggal future yang telah ditentukan sebelumnya. Untuk informasi lebih lanjut, lihat [Membuat perjanjian bertanggal future untuk penawaran pribadi](#).

Bekerja dengan perjanjian

Anda melihat dan mengelola perjanjian dari halaman Perjanjian di [Portal Manajemen AWS Marketplace](#). Pada halaman Perjanjian, tabel Perjanjian menunjukkan kontrak untuk produk yang Anda jual AWS Marketplace. Dalam tabel Perjanjian, perjanjian dapat memiliki salah satu status berikut:

- Aktif — Ketentuan perjanjian aktif.
- Kedaluwarsa — Kontrak akan berakhir dalam waktu kurang dari 90 hari.
- Kedaluwarsa — Perjanjian berakhir pada tanggal akhir kontrak yang ditentukan.

- Diperpanjang - Perjanjian diperbarui, menciptakan perjanjian baru dengan persyaratan yang diperbarui.
- Dibatalkan — Akseptor mengakhiri perjanjian lebih awal.
- Diganti - Perjanjian diganti menggunakan penawaran penggantian perjanjian.
- Diakhiri — Perjanjian diakhiri AWS sebelum tanggal akhir kontrak asli karena alasan seperti kegagalan pembayaran.
- Diarsipkan — Perjanjian berakhir tanpa alasan tertentu.

Menemukan perjanjian

Untuk menemukan kesepakatan, atur filter. Pilih bilah pencarian Temukan perjanjian, lalu pilih properti dan nilai filter. Misalnya, untuk menemukan perjanjian tertentu, pilih ID Perjanjian, lalu pilih ID. Untuk menyimpan filter, untuk Hapus filter, pilih Simpan sebagai set filter baru.

Menyortir perjanjian

Tabel awalnya diurutkan berdasarkan kolom Perjanjian berakhir pada (UTC). Ini menampilkan perjanjian dalam urutan menurun, dengan yang berakhir terjauh di masa depan muncul di bagian atas daftar. Anda dapat memilih judul kolom Perjanjian berakhir pada (UTC) untuk mengubah urutan pengurutan.

Melihat detail perjanjian

Untuk melihat detail perjanjian, pilih tautan di kolom ID Perjanjian. Atau, Anda dapat memilih opsi di sebelah perjanjian dan memilih Lihat detail.

Note

Untuk mengubah perjanjian, lihat. [Mengubah perjanjian di AWS Marketplace](#)

Mengunduh detail perjanjian

Gunakan prosedur berikut untuk mengunduh detail perjanjian dalam file.pdf.

Untuk mengunduh detail perjanjian

1. Masuk ke [Portal Manajemen AWS Marketplace](#), dan pilih Perjanjian.

2. Dalam tabel Perjanjian, pilih opsi di sebelah perjanjian dan pilih Lihat detail. Atau, Anda dapat memilih tautan untuk perjanjian di kolom ID Perjanjian.
3. Pada halaman detail perjanjian, pilih Unduh PDF.

Mengubah perjanjian di AWS Marketplace

AWS Marketplace penjual, Mitra Saluran, dan ISVs dapat menawarkan peningkatan, pembaruan, dan amandemen untuk menggantikan perjanjian aktif. [Misalnya, Anda dapat memberikan hak baru, menawarkan diskon harga, menyesuaikan jadwal pembayaran, atau mengubah perjanjian lisensi pengguna akhir \(EULA\) untuk menggunakan persyaratan lisensi standar.](#) Anda juga dapat mengubah jumlah unit dan jadwal pembayaran dan menambahkan tanggal akhir kustom.

Kontrak perangkat lunak sebagai layanan (SaaS) dan kontrak SaaS dengan produk konsumsi mendukung amandemen penawaran publik dan swasta. Semua AWS Marketplace penjual dapat meningkatkan, memperbarui, atau mengubah penawaran pribadi untuk jenis produk ini, termasuk vendor perangkat lunak independen (ISVs) dan mitra saluran. Bagian berikut memberikan informasi tentang proses tersebut.

Perbedaan antara penawaran dan perjanjian adalah apakah pembeli menerima persyaratannya:

- Penawaran adalah seperangkat persyaratan untuk penggunaan produk oleh pembeli. Penawaran dapat bersifat publik atau pribadi.
- Perjanjian adalah penawaran yang diterima pembeli. Perjanjian termasuk produk yang dibeli dan gratis yang disediakan penjual menggunakan penawaran publik atau pribadi.

Anda tidak dapat mengubah perjanjian untuk menentukan penjual catatan yang berbeda dari penjual catatan dari perjanjian asli. Untuk menggunakan fitur ini, Anda harus memiliki izin untuk menggunakan tab Perjanjian di Portal Manajemen AWS Marketplace Untuk informasi, lihat [Izin untuk penjual AWS Marketplace](#).

Topik

- [Jenis produk yang didukung untuk amandemen penawaran publik dan pribadi](#)
- [Membuat peningkatan, pembaruan, dan amandemen penawaran publik dan pribadi](#)
- [Pelaporan untuk upgrade, perpanjangan, dan amandemen](#)

Jenis produk yang didukung untuk amandemen penawaran publik dan pribadi

Hanya jenis produk berikut yang mendukung amandemen penawaran:

- Kontrak SaaS
- Kontrak SaaS dengan konsumsi

Anda dapat melihat jenis produk tambahan berikut di tab Perjanjian di Portal Manajemen AWS Marketplace. Namun, jenis produk ini tidak mendukung amandemen:

- Produk berbasis penggunaan SaaS
- Produk berbasis AMI
- Produk berbasis kontainer
- Kontrak server
- Produk layanan profesional

Membuat peningkatan, pembaruan, dan amandemen penawaran publik dan pribadi

Anda dapat membuat upgrade penawaran, perpanjangan, dan amandemen dari menggunakan prosedur berikut. Portal Manajemen AWS Marketplace Untuk penawaran pribadi Mitra Saluran, Mitra Saluran harus menggunakan mata uang yang ditentukan dalam otorisasi penjualan saat membuat amandemen.

Note

- Penawaran yang diubah tidak dapat memiliki tanggal mulai perjanjian masa depan. Penawaran yang diubah hanya menargetkan perjanjian aktif, dan setelah diterima, perjanjian yang diubah segera menjadi aktif membuat kencana masa depan tidak mungkin.
- Jika Anda mengubah penawaran publik yang diterima, itu menjadi penawaran pribadi dan tidak lagi diperpanjang secara otomatis. Untuk mempertahankan pembaruan otomatis, pembeli harus berlangganan penawaran umum. Untuk informasi lebih lanjut, lihat [Bagaimana penawaran pribadi bekerja](#).

Untuk membuat upgrade penawaran, perpanjangan, dan amandemen

1. Masuk ke [Portal Manajemen AWS Marketplace](#) dan pilih Perjanjian.
2. Pada halaman Perjanjian, pilih kotak centang di samping perjanjian, lalu pilih Lihat Detail.
3. Pada halaman Lihat perjanjian, pilih Ubah perjanjian.
4. Pada halaman Amendemen detail perjanjian, Anda juga dapat membuat perubahan pada tanggal layanan, dimensi produk, mata uang penawaran (untuk PO langsung AWS MP), jadwal pembayaran, dimensi penggunaan, status perpanjangan, EULA, dan tanggal kedaluwarsa penawaran.
5. Tinjau penawaran dan pilih Buat penawaran.

 Tip

Memasukkan nama penawaran khusus deskriptif dapat membantu Anda membedakan antara penawaran aktif Anda di halaman Penawaran. Nama penawaran khusus juga terlihat oleh pembeli.

AWS merekomendasikan menggunakan nama penawaran khusus yang mencakup rincian identifikasi tambahan, seperti nomor pesanan Anda sendiri IDs dan pembelian. Menggunakan deskripsi tingkat tinggi seperti **upgrade** atau **renewal** dan nama perusahaan khusus juga direkomendasikan. Jangan gunakan data yang dapat diidentifikasi secara pribadi (misalnya, nama depan atau belakang, nomor telepon, atau alamat). Anda dapat memasukkan hingga 150 karakter untuk bidang ini.

Lebih lanjut tentang penawaran yang diubah

Penawaran yang diubah akan muncul di halaman Penawaran Pribadi dalam waktu sekitar 45 menit. Untuk melihat penawaran, masuk ke Portal Manajemen AWS Marketplace dan pilih Penawaran. Pada halaman Penawaran Pribadi, pembeli memiliki opsi untuk menerima penawaran atau melanjutkan perjanjian asli.

Jika pembeli menerima peningkatan atau pembaruan penawaran publik atau pribadi, perjanjian baru segera berlaku dan perjanjian tersebut tercantum di halaman Perjanjian di Portal Manajemen AWS Marketplace. Pembayaran terjadwal yang tersisa dari perjanjian sebelumnya dibatalkan. Pembeli menerima amandemen dengan cara yang sama mereka menerima penawaran pribadi. Untuk informasi selengkapnya tentang pengalaman pembeli untuk penawaran pribadi, lihat [Penawaran pribadi](#) di Panduan AWS Marketplace Pembeli.

Jika pembeli tidak menerima peningkatan atau pembaruan penawaran publik atau pribadi sebelum kedaluwarsa, perjanjian asli akan terus tidak berubah.

 Note

Untuk [notifikasi Amazon SNS untuk produk SaaS](#), subscribe-success pesan dikirim dengan yang baru offer-identifier ketika pembeli menerima amandemen.

Pelaporan untuk upgrade, perpanjangan, dan amandemen

Peningkatan dan pembaruan penawaran publik dan pribadi muncul di laporan penjual yang ada dan dalam laporan yang relevan dengan penawaran. Itu [Laporan pelanggan pelanggan harian](#) dan [Laporan bisnis harian](#) dihasilkan setiap hari. [Laporan pendapatan bulanan yang ditagih](#) itu dihasilkan setiap bulan.

Dalam laporan pelanggan harian, bidang Maksud berlangganan menunjukkan apakah entri laporan adalah penawaran pribadi baru. Kolom ID penawaran sebelumnya menampilkan ID dari penawaran sebelumnya. Semua penawaran pribadi diberi label sebagai “pribadi” dalam laporan.

 Important

Amandemen menggantikan langganan pembeli saat ini. Faktur yang ada tetap tidak berubah. Namun, jadwal pembayaran dalam amandemen menggantikan faktur yang tertunda dari langganan sebelumnya.

Produk berbasis AMI di AWS Marketplace

Sebagai AWS Marketplace penjual, Anda dapat mengirimkan produk Anda ke pembeli dengan [Amazon Machine Images \(AMIs\)](#). AMI menyediakan informasi yang diperlukan untuk meluncurkan instans Amazon Elastic Compute Cloud (Amazon EC2). Anda membuat AMI khusus untuk produk Anda, dan pembeli dapat menggunakannya untuk membuat EC2 instance dengan produk Anda yang sudah diinstal dan siap digunakan. Topik ini memberikan informasi tentang penggunaan AMIs untuk mengirimkan AWS Marketplace produk Anda.

Saat pembeli menggunakan AMI yang Anda berikan, mereka ditagih untuk instance yang mereka buat, mengikuti opsi harga dan pengukuran yang Anda buat untuk produk Anda. Pembeli dapat menggunakan AMI produk Anda dengan cara yang sama seperti yang mereka gunakan AMIs di lain AWS, termasuk membuat AMI versi kustom baru. EC2 instance yang dibuat dari AMI masih ditagih sebagai produk Anda, berdasarkan kode produk AMI.

Metode pengantaran produk berbasis AMI

Anda dapat mengirimkan produk berbasis AMI dengan salah satu cara berikut:

- AMI Tunggal - Pembeli memilih dan menggunakan AMI sebagai template untuk sebuah EC2 contoh. Pembeli dapat menemukan produk ini menggunakan filter metode pengiriman Amazon Machine Image. Untuk informasi selengkapnya, lihat [Membuat produk berbasis AMI](#).
- AWS CloudFormation template — Anda membuat template yang memungkinkan pembeli untuk menginstal sistem beberapa instance dengan peran yang berbeda sebagai satu unit. Pembeli dapat menemukan produk ini menggunakan filter metode CloudFormation pengiriman. Untuk informasi selengkapnya, lihat [Tambahkan CloudFormation template ke produk Anda](#).

Sumber daya tambahan

Untuk informasi selengkapnya tentang produk AMI, lihat topik berikut.

AWS Marketplace

- [Harga produk untuk AWS Marketplace](#)
- [Mengkonfigurasi pengukuran khusus untuk produk AMI dengan AWS Marketplace Metering Service](#)

Produk berbasis AMI

- [Memahami produk berbasis AMI di AWS Marketplace](#)
- [Membuat produk berbasis AMI](#)
- [Mengelola produk berbasis AMI sebagai penjual AWS Marketplace](#)
- [Tambahkan CloudFormation template ke produk Anda](#)
- [Praktik terbaik untuk membangun AMIs untuk digunakan dengan AWS Marketplace](#)
- [Harga produk AMI untuk AWS Marketplace](#)
- [Menerima notifikasi Amazon SNS untuk produk AMI AWS Marketplace](#)
- [Daftar periksa produk AMI untuk AWS Marketplace](#)
- [Persyaratan produk berbasis AMI untuk AWS Marketplace](#)

Memahami produk berbasis AMI di AWS Marketplace

Sebagai AWS Marketplace penjual, Anda dapat mengirimkan produk Anda ke pembeli dengan [Amazon Machine Images \(AMIs\)](#). AMI menyediakan informasi yang diperlukan untuk meluncurkan instans Amazon Elastic Compute Cloud (Amazon EC2). Bagian berikut menjelaskan konsep kunci untuk bekerja dengan produk berbasis AMI.

Note

Anda hanya dapat menggunakan satu AMI dalam produk AMI, tetapi Anda dapat menambahkan versi AMI tersebut ke produk.

Topik

- [Siklus hidup produk](#)
- [Kode produk AMI](#)
- [Perubahan permintaan](#)
- [Formulir Beban Produk](#)
- [Amandemen perjanjian tahunan](#)

Siklus hidup produk

Produk berbasis AMI mencakup satu atau lebih versi perangkat lunak, ditambah metadata tentang produk secara keseluruhan. Saat membuat produk, Anda mengonfigurasi propertinya AWS Marketplace termasuk nama produk, deskripsi, dan harga. Anda juga menentukan kategori yang sesuai untuk produk Anda dan menambahkan kata kunci sehingga produk Anda muncul dalam pencarian yang relevan.

Anda juga membuat versi pertama perangkat lunak. Ini bisa berupa AMI, atau Anda dapat menyertakan AWS CloudFormation templat atau paket perangkat lunak yang dapat digunakan pembeli untuk membuatnya sendiri AMIs. Untuk informasi selengkapnya, lihat [Metode pengantaran produk berbasis AMI](#).

Untuk produk berbayar, pembeli ditagih untuk jumlah instans yang terinstal. Untuk mengukur dimensi berbeda yang dilacak perangkat lunak Anda, seperti jumlah pengguna produk, integrasikan produk Anda dengan Layanan AWS Marketplace Pengukuran. Untuk informasi selengkapnya, lihat [Mengonfigurasi pengukuran khusus untuk produk AMI dengan AWS Marketplace Metering Service](#).

Saat Anda membuat produk dan versi pertama perangkat lunak, produk ini awalnya diterbitkan dalam lingkup terbatas sehingga hanya akun Anda yang dapat mengaksesnya. Ketika Anda siap, Anda dapat mempublikasikannya ke AWS Marketplace katalog untuk memungkinkan pembeli berlangganan dan membeli produk Anda.

Anda menggunakan halaman [produk Server](#) untuk melihat daftar produk Anda. Produk akan memiliki salah satu status berikut:

- **Pementasan** — Produk yang tidak lengkap yang masih Anda tambahkan informasinya. Pada Simpan dan keluar pertama dari pengalaman membuat layanan mandiri, permintaan perubahan yang berhasil membuat produk yang tidak dipublikasikan dengan informasi dari langkah-langkah lengkap yang Anda kirimkan. Dari keadaan ini, Anda dapat terus menambahkan informasi ke produk atau mengubah detail yang sudah dikirimkan melalui permintaan perubahan.
- **Terbatas** — Sebuah produk selesai setelah diserahkan ke sistem dan melewati semua validasi dalam sistem. Kemudian produk dirilis ke keadaan Terbatas. Pada titik ini, produk memiliki halaman detail yang hanya dapat diakses oleh akun Anda dan siapa pun yang Anda izinkan. Anda dapat menguji produk Anda melalui halaman detail. Jika perlu, untuk informasi dan bantuan lebih lanjut, hubungi [tim Operasi AWS Marketplace Penjual](#).
- **Publik** — Saat Anda siap mempublikasikan produk sehingga pembeli dapat melihat dan berlangganan produk, Anda menggunakan permintaan perubahan visibilitas Perbarui. [Ini memulai](#)

[alur kerja untuk tim Operasi AWS Marketplace Penjual untuk meninjau dan mengaudit produk Anda terhadap kebijakan kami.](#) Setelah produk disetujui dan permintaan perubahan diproses, produk dipindahkan dari status Terbatas ke Publik.

- Dibatasi - Jika Anda ingin menghentikan pengguna baru berlangganan produk Anda, Anda dapat membatasi produk dengan menggunakan permintaan perubahan visibilitas Perbarui. Status Terbatas berarti bahwa pengguna yang ada dapat terus menggunakan produk. Namun, produk tidak akan lagi terlihat oleh publik atau tersedia untuk pengguna baru.

Siklus hidup produk berbasis AMI untuk AWS Marketplace tidak berakhir setelah Anda mempublikasikan versi pertama. Anda harus selalu memperbarui produk Anda dengan versi baru perangkat lunak Anda, dan dengan tambalan keamanan untuk sistem operasi dasar.

Sebagai contoh siklus hidup produk berbasis AMI lengkap, bayangkan penjual ingin menjual produk berbasis AMI mereka di AWS Marketplace. Berikut adalah bagaimana penjual membuat dan memelihara produk dari waktu ke waktu:

1. Buat produk — Penjual membuat produk, dan menerbitkan versi 1.0.0 ke AWS Marketplace. Pembeli dapat membuat instans versi 1.0.0 dan menggunakannya.
2. Menambah versi baru – Kemudian, penjual menambahkan fitur baru untuk produk, dan menambahkan versi baru, 1.1.0, yang menyertakan fitur tersebut. Pembeli masih dapat menggunakan versi asli, 1.0.0, atau mereka dapat memilih versi baru, 1.1.0.

Note

Tidak seperti produk baru, versi baru dipublikasikan untuk sepenuhnya tersedia untuk publik. Anda hanya dapat mengujinya AWS Marketplace tanpa pelanggan melihatnya jika produk secara keseluruhan dalam rilis terbatas.

3. Memperbarui informasi produk – Dengan versi 1.1.0 yang tersedia, penjual memungkinkan pembeli mengetahui berbagai fitur baru dengan memperbarui informasi produk dengan teks sorotan baru yang menguraikan fitur tersebut.
4. Menambahkan versi minor – Saat penjual memperbaiki bug di versi 1.1.0, mereka merilis perbaikan tersebut dengan menambahkan versi baru 1.1.1. Pembeli sekarang memiliki pilihan untuk menggunakan versi 1.0.0, 1.1.0, atau 1.1.1.
5. Membatasi versi – Penjual memutuskan bahwa bug cukup serius sehingga mereka tidak ingin pembeli dapat menggunakan versi 1.1.0, sehingga mereka membatasi versi tersebut. Tidak ada

- pelanggan baru yang kemudian dapat membeli 1.1.0 (mereka hanya dapat memilih 1.0.0 atau 1.1.1), meski pembeli yang ada masih memiliki akses ke versi tersebut.
6. Memperbarui informasi versi – Untuk membantu pembeli yang sudah ada, penjual memperbarui informasi versi untuk 1.1.0 dengan saran untuk memutakhirkan ke versi 1.1.1.
 7. Pantau penggunaan — Saat pembeli membeli dan menggunakan produk, penjual memantau penjualan, penggunaan, dan metrik lainnya menggunakan. AWS Marketplace [Laporan penjual, umpan data, dan dasbor di AWS Marketplace](#)
 8. Hapus produk — Ketika produk tidak lagi dibutuhkan, penjual menghapusnya AWS Marketplace.

Dalam contoh ini, penjual membuat tiga versi AMI yang berbeda dalam produk, tetapi hanya dua yang tersedia untuk pembeli baru (sebelum menghapus produk).

Untuk membuat modifikasi pada versi atau informasi produk, Anda membuat [permintaan perubahan](#) di Portal Manajemen AWS Marketplace.

Untuk petunjuk terperinci tentang langkah-langkah untuk membuat dan mengelola produk berbasis AMI, lihat [Membuat produk berbasis AMI](#).

Kode produk AMI

Kode produk unik ditetapkan ke produk Anda saat Anda membuatnya di AWS Marketplace. Kode produk tersebut dikaitkan dengan AMI untuk produk Anda dan digunakan untuk melacak penggunaan produk Anda. Kode produk di-deploy secara otomatis saat pembeli bekerja dengan perangkat lunak. Misalnya, pelanggan berlangganan dan meluncurkan AMI, mengonfigurasinya, dan menghasilkan AMI baru. AMI baru masih berisi kode produk asli, sehingga pelacakan penggunaan dan izin yang benar tetap ada.

Note

Kode produk berbeda dari ID produk untuk produk Anda. Setiap produk di AWS Marketplace diberi ID produk unik. ID produk digunakan untuk mengidentifikasi produk Anda di katalog AWS Marketplace, dalam penagihan pelanggan, dan dalam laporan penjual. Kode produk dilampirkan ke instans yang dibuat dari AMI Anda sebagai metadata instans. Saat AMI dengan kode produk yang digunakan untuk membuat sebuah instans, pelanggan akan mendapatkan tagihan yang menunjukkan ID produk terkait. Setelah Anda membuat produk Anda, temukan kode produk dan ID produk di Portal Manajemen AWS Marketplace halaman untuk produk Anda.

Sebagai penjual, perangkat lunak Anda bisa mendapatkan kode produk untuk instans Amazon Elastic Compute Cloud (Amazon EC2) yang sedang berjalan saat runtime dari metadata instans. Anda dapat menggunakan kode produk untuk keamanan tambahan, seperti memvalidasi kode produk pada awal produk. Anda tidak dapat membuat panggilan API ke kode produk AMI sampai produk telah dipublikasikan dalam status terbatas untuk pengujian. Untuk informasi selengkapnya tentang memverifikasi kode produk, lihat [Memverifikasi perangkat lunak Anda berjalan di perangkat AWS Marketplace AMI](#).

Perubahan permintaan

Untuk membuat perubahan pada produk atau versi AWS Marketplace, Anda mengirimkan permintaan perubahan melalui Portal Manajemen AWS Marketplace. Permintaan perubahan ditambahkan ke antrian dan dapat memakan waktu dari hitungan menit hingga hitungan untuk diselesaikan, tergantung tipe permintaannya. Anda dapat melihat status permintaan di Portal Manajemen AWS Marketplace.

Note

Selain itu Portal Manajemen AWS Marketplace, Anda juga dapat membuat permintaan perubahan dengan menggunakan [API AWS Marketplace Katalog](#).

Tipe perubahan yang dapat Anda minta untuk produk berbasis AMI meliputi:

- Memperbarui informasi produk yang ditampilkan kepada pembeli.
- Memperbarui informasi versi yang ditampilkan kepada pembeli.
- Menambahkan versi baru produk Anda.
- Membatasi versi sehingga pembeli baru tidak dapat lagi mengakses versi tersebut.
- Wilayah AWS Perbarui tempat produk tersedia.
- Memperbarui harga dan contoh tipe untuk produk.
- Hapus produk dari AWS Marketplace.

Note

Beberapa permintaan perubahan mengharuskan Anda untuk menggunakan formulir beban produk untuk membuat permintaan. Untuk informasi selengkapnya, lihat [Formulir Beban Produk](#).

Perbarui permintaan perubahan

Ubah permintaan yang dimulai dengan pembaruan akan memuat detail saat ini. Anda kemudian membuat pembaruan, yang menimpa detail yang ada.

Tambahkan atau batasi permintaan perubahan

Tambah dan batasi pasangan permintaan khusus untuk langkah-langkah dan pembaruan yang disediakan setelah setiap permintaan berhasil. Permintaan berhasil setelah Anda memilih Simpan dan keluar dan Kirim tindakan dalam pengalaman layanan mandiri.

Misalnya, jika aset AMI disediakan ke instans dan Wilayah setelah ditambahkan, maka aset tersebut hanya dapat dibatasi daripada dihapus sepenuhnya. Ini berarti pelanggan dan pengguna yang ada dapat terus menggunakan produk sampai langganan atau kontrak mereka habis. Namun, tidak ada pelanggan baru yang dapat ditambahkan ke produk yang berada dalam status Terbatas.

Formulir Beban Produk

Biasanya, Anda gunakan [Portal Manajemen AWS Marketplace](#) untuk membuat atau mengedit produk Anda. Namun, beberapa operasi mengharuskan Anda untuk menggunakan Formulir Beban Produk (PLF).

Sebuah PLF adalah spreadsheet yang berisi semua informasi tentang produk. Untuk mendapatkan PLF, Anda dapat:

- Unduh PLF untuk produk yang sudah ada dari halaman detail produk di Portal Manajemen AWS Marketplace
- Pilih item menu untuk tindakan yang membutuhkan PLF. Misalnya, jika Anda membuat produk server tagihan bulanan baru, sistem meminta Anda untuk mengunduh PLF yang sesuai.

Jika tindakannya adalah pengeditan ke produk yang sudah ada, PLF sudah diisi sebelumnya dengan informasi untuk produk tersebut, jadi Anda hanya perlu mengubah detail yang ingin Anda perbarui.

- Jika Anda membutuhkan PLF baru yang kosong, navigasikan ke halaman [Unggah Portal Manajemen AWS Marketplace file](#). Halaman berisi tautan PLFs untuk jenis produk yang ingin Anda buat.

Setelah Anda menyelesaikan PLF Anda, unggah ke halaman [unggah Portal Manajemen AWS Marketplace File](#). PLF sendiri mempunyai petunjuk yang lebih terperinci dalam tab Petunjuk.

Amandemen perjanjian tahunan

Amandemen rencana tahunan (tahunan) per jam memungkinkan Anda dan pembeli Anda untuk membuat perubahan berikut pada paket yang ada:

- Beralih di antara keluarga tipe EC2 instans Amazon
- Beralih di antara ukuran jenis EC2 instans Amazon
- Tambahkan jenis instance baru
- Tingkatkan jumlah jenis instans yang ada dalam perjanjian

Pembeli dapat melakukan perubahan selama biaya perubahan yang diprorata lebih besar dari nol (mereka tidak dapat menurunkan nilai langganan). Biaya prorata dari EC2 instans Amazon yang baru ditambahkan didasarkan pada biaya tahunan dari jenis instans yang disesuaikan dengan jangka waktu perjanjian yang tersisa. Saat mengganti jenis instans, biaya prorata dari jenis EC2 instans Amazon yang dihapus dikurangkan dari biaya prorata jenis instans Amazon yang baru ditambahkan. EC2

Tidak diperlukan tindakan tambahan untuk mengaktifkan amandemen pada produk tahunan AMI. Amandemen didukung pada semua perjanjian yang dibuat dari penawaran publik dan perjanjian dari penawaran pribadi yang tidak menggunakan paket angsuran.

Anda dapat melihat amandemen yang dibuat oleh pembeli Anda di dasbor berikut:

- [Perjanjian dan pembaruan dasboard](#) — Daftar perjanjian yang diubah.
- [Dasbor pendapatan yang ditagih](#) — Biaya kepada pelanggan.
- [Dasbor koleksi dan pencairan — Pencairan](#).

Membuat produk berbasis AMI

Pengalaman layanan mandiri Amazon Machine Image (AMI) memandu Anda saat membuat daftar produk dan membuat permintaan perubahan. Dengan menggunakan pengalaman swalayan, Anda dapat memperbarui daftar produk Anda secara langsung dengan lebih sedikit waktu yang dibutuhkan untuk diproses oleh tim Operasi AWS Marketplace Penjual. Banyak langkah pengalaman swalayan selaras dengan sistem katalog di AWS Marketplace, yang memfasilitasi validasi langsung alih-alih menunggu pemrosesan dan validasi dari tim Operasi Penjual. AWS Marketplace Topik ini menjelaskan cara menggunakan pengalaman swalayan AMI untuk membuat daftar produk untuk satu AMI. Pelanggan gunakan AMIs untuk membuat EC2 instans Amazon dengan produk Anda yang sudah diinstal dan dikonfigurasi.

Topik

- [Prasyarat](#)
- [Memahami pengalaman swalayan](#)
- [Buat daftar](#)
- [Sumber daya tambahan](#)

Prasyarat

Sebelum Anda membuat daftar produk AMI, Anda harus melengkapi prasyarat berikut:

1. Memiliki akses ke Portal Manajemen AWS Marketplace. Ini adalah alat yang Anda gunakan untuk mendaftar sebagai penjual dan mengelola produk yang Anda jual AWS Marketplace. Untuk mempelajari lebih lanjut tentang mendapatkan akses ke Portal Manajemen AWS Marketplace, lihat [Kebijakan dan izin untuk penjual AWS Marketplace](#).
2. Mendaftar sebagai penjual dan, jika Anda ingin mengenakan biaya untuk produk Anda, kirimkan pajak dan informasi perbankan Anda. Untuk mempelajari selengkapnya tentang menjadi penjual, lihat [Memulai sebagai AWS Marketplace penjual](#).
3. Memiliki produk yang ingin Anda jual. Untuk produk berbasis AMI, ini biasanya berarti Anda telah membuat atau memodifikasi perangkat lunak server Anda, dan membuat AMI untuk digunakan pelanggan Anda. Untuk mempelajari lebih lanjut tentang menyiapkan AMI untuk digunakan AWS Marketplace, lihat [Praktik terbaik untuk membangun AMIs untuk digunakan dengan AWS Marketplace](#).

Memahami pengalaman swalayan

Pengalaman swalayan memandu Anda melalui pembuatan produk Anda. AWS Marketplace Saat Anda melanjutkan langkah-langkahnya, Anda menentukan informasi produk dan setelan penerapan AMI, seperti Wilayah AWS, jenis instans, dan detail AMI. Anda juga mengonfigurasi detail transaksi termasuk harga, ketersediaan negara, EULA, dan kebijakan pengembalian dana. Sebagai opsi, Anda dapat menentukan daftar yang diizinkan Akun AWS IDs untuk mengakses dan menguji produk saat berada dalam status Terbatas.

Sebelum Anda mulai, tinjau aspek-aspek kunci berikut dari pengalaman swalayan:

- Anda hanya dapat pergi ke langkah berikutnya setelah Anda menyelesaikan bidang yang diperlukan pada langkah saat ini. Persyaratan ini karena ada validasi tingkat halaman di akhir setiap langkah. Anda tidak dapat menyimpan atau mengirimkan langkah yang tidak lengkap.
- Jika Anda perlu mengakhiri sesi sebelum menyelesaikan semua langkah dalam proses, Anda dapat memilih Simpan dan keluar untuk mengirimkan langkah-langkah yang Anda selesaikan ke area pementasan.
- Langkah yang tidak lengkap dan tidak lulus validasi tidak dikirimkan ke sistem. Langkah yang diselesaikan sebagian tidak valid dan tidak dapat disimpan.
- Saat Anda memilih Simpan dan keluar, kotak dialog Simpan dan keluar menunjukkan langkah-langkah yang melewati pemeriksaan validasi. Anda dapat meninjau dan memilih untuk menyimpan hingga langkah-langkah terakhir yang telah diselesaikan dan divalidasi. Jika ada kesalahan validasi atau detail yang hilang, Anda dapat memilih Perbaiki untuk kembali ke langkah itu.
- Setelah Anda Menyimpan dan keluar, permintaan sedang ditinjau saat sedang diproses. Ini bisa memakan waktu beberapa menit atau jam untuk menyelesaikan pemrosesan. Anda tidak dapat melanjutkan langkah-langkah atau membuat perubahan sampai permintaan berhasil. Untuk Simpan dan keluar pertama, permintaan membuat produk secara paralel dengan langkah-langkah yang telah Anda selesaikan.
- Setelah permintaan berhasil, Anda telah menyelesaikan penyimpanan. Untuk melanjutkan perubahan pada halaman ikhtisar Produk, pilih Lanjutkan pembuatan produk atau gunakan Minta perubahan untuk memperbarui detail yang telah Anda kirimkan sebelumnya di sesi terakhir. Saat Anda melanjutkan, perhatikan bahwa langkah-langkah yang telah Anda selesaikan ditandai dengan label Succeeded hijau. Untuk memperbarui langkah yang dikirimkan sebelumnya, gunakan Permintaan perubahan. Permintaan Simpan dan keluar sebelumnya harus diselesaikan sebelum Anda dapat melanjutkan langkah ini.

- Setelah menyelesaikan semua langkah, Anda dapat memilih Berikutnya untuk melihat ulasan. Pilih Kirim permintaan mana yang sistem melakukan validasi akhir. Setelah Anda menerima respons Berhasil, produk berpindah ke status Terbatas. Anda dapat melihat di halaman detail bahwa produk sekarang tersedia untuk siapa saja di daftar yang diizinkan. Jika permintaan gagal, produk tetap dalam status Pementasan dan mengharuskan Anda untuk melakukan koreksi sebelum mengirimkan ulang.

Buat daftar

Langkah-langkah di bagian ini menjelaskan cara membuat daftar untuk produk Single-AMI.

Note

Anda hanya dapat pergi ke langkah berikutnya ketika Anda menyelesaikan bidang yang diperlukan pada langkah saat ini. Anda tidak dapat menyimpan atau mengirimkan langkah yang tidak lengkap. Jika Anda perlu mengakhiri sesi sebelum menyelesaikan semua langkah dalam proses, pilih Simpan dan keluar untuk mengirimkan langkah-langkah yang Anda selesaikan ke area pementasan. Untuk informasi selengkapnya, lihat [Memahami pengalaman swalayan](#).

Untuk membuat produk AMI tunggal

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/>, lalu masuk ke akun penjual Anda.
2. Dari menu Produk, pilih Server. Atau, Anda bisa langsung menuju halaman [Produk Server](#).
3. Dari tab Produk Server, pilih Buat produk server, pilih Amazon Machine Image (AMI), lalu pilih salah satu jenis lisensi untuk produk Single-AMI:
 - Bawa lisensi Anda sendiri (BYOL) — Produk yang pengguna mendapat lisensi dari Anda di luar. AWS Marketplace Ini bisa berupa lisensi berbayar atau gratis.
 - Gratis – Produk yang dapat digunakan oleh pelanggan Anda secara gratis. (Mereka masih akan membayar biaya untuk instans Amazon Elastic Compute Cloud (Amazon EC2) terkait, atau AWS sumber daya lainnya.)
 - Dibayar per jam atau per jam tahunan — Produk yang dibayar pembeli baik setiap jam atau per jam dengan kontrak tahunan. AWS melakukan pengukuran berdasarkan kode produk pada AMI.

- Dibayar bulanan – Produk yang ditagihkan kepada pembeli secara bulanan oleh AWS.
 - Penggunaan berbayar — Perangkat lunak dibebankan langsung untuk nilai yang Anda berikan, bersama dengan salah satu dari empat kategori penggunaan: pengguna, data, bandwidth, atau host. Anda dapat menentukan hingga 24 dimensi untuk produk. Semua biaya masih dikeluarkan oleh pelanggan.
 - AMI dengan harga kontrak — Produk Single-AMI atau Single-AMI dengan AWS CloudFormation tumpukan yang pembeli membayar biaya di muka.
4. Pengalaman swalayan memandu Anda melalui langkah-langkah untuk membuat AWS Marketplace daftar. Anda harus memasukkan informasi produk (metadata), detail penyebaran produk (, contoh Wilayah AWS, dan detail AMI), dan detail penawaran publik (harga, EULA, ketersediaan berdasarkan negara, EULA, pengembalian dana). Sebagai opsi, Anda dapat menambahkan akun ke daftar yang diizinkan untuk menguji produk. Selesaikan setiap langkah untuk pindah ke langkah berikutnya dalam proses.

 Note

Jika Anda perlu mengakhiri sesi sebelum menyelesaikan semua langkah dalam proses, Anda dapat menggunakan fungsi Simpan dan keluar untuk mengirimkan langkah-langkah yang Anda selesaikan ke area pementasan. Ini menciptakan permintaan untuk informasi yang Anda berikan untuk divalidasi. Saat permintaan sedang diproses, Anda tidak dapat mengedit produk. Setelah permintaan berhasil, Anda dapat terus membuat produk Anda dengan memilih Lanjutkan pembuatan produk.

Permintaan yang gagal berarti tidak ada pembaruan yang dilakukan pada produk karena kesalahan validasi. Ini akan terlihat pada log permintaan untuk produk Anda. Anda dapat memilih permintaan untuk melihat kesalahan, menggunakan Salin ke baru di bawah Tindakan untuk memperbaiki kesalahan, dan mengirim ulang permintaan. Saat Anda melanjutkan langkah-langkahnya, Anda dapat melanjutkan dari langkah setelah langkah yang Anda simpan di sesi terakhir. Untuk memperbarui langkah-langkah sebelumnya, buka halaman ikhtisar produk dan kirimkan permintaan perubahan untuk memperbarui langkah-langkah yang Anda kirimkan sebelumnya.

5. Setelah memasukkan informasi yang diperlukan untuk semua langkah permintaan perubahan, pilih Kirim. Pengajuan ini membuat permintaan ke sistem AWS Marketplace katalog untuk memvalidasi informasi dan melepaskan produk ke status Terbatas, jika validasi lolos. Saat permintaan sedang diproses, Anda tidak dapat terus mengedit produk. Setelah permintaan berhasil, produk dipindahkan ke keadaan Terbatas.

- Ketika produk Anda awalnya diterbitkan, itu hanya dapat diakses oleh Anda Akun AWS (yang Anda gunakan untuk membuat produk) dan akun pengujian tim Operasi AWS Marketplace Penjual. Jika Anda melihat produk dari halaman produk Server, Anda dapat memilih Lihat AWS Marketplace untuk melihat detail produk karena akan muncul AWS Marketplace untuk pembeli. Daftar detail ini tidak terlihat oleh AWS Marketplace pengguna lain.
 - Kemampuan ini memungkinkan Anda untuk menguji produk Anda (dan bahkan mempublikasikan beberapa versi untuk pengujian) sebelum merilisnya ke publik.
6. Uji produk Anda dalam keadaan Terbatas dan pastikan produk tersebut mengikuti [persyaratan produk AWS Marketplace berbasis AMI dan daftar](#) periksa [produk](#). Kemudian, untuk meminta agar produk Anda dipublikasikan ke Publik, pilih Perbarui visibilitas. Tim Operasi AWS Marketplace Penjual harus meninjau produk Anda sebelum menyetujuinya untuk go Public.

Note

Verifikasi dan publikasi produk adalah proses manual, yang ditangani oleh tim Operasi AWS Marketplace Penjual. Diperlukan waktu 7-10 hari kerja untuk mempublikasikan versi produk awal Anda, jika tidak ada kesalahan. Untuk detail selengkapnya tentang pengaturan waktu, lihat [Pengaturan waktu dan harapan](#).

Untuk informasi selengkapnya tentang menyiapkan dan mengirimkan produk Single-AMI dan informasi produk Anda, lihat. [Sumber daya tambahan](#)

Sumber daya tambahan

Untuk informasi selengkapnya tentang mempersiapkan informasi produk Anda dan mengirimkannya untuk publikasi, lihat sumber daya berikut:

- [Mempersiapkan produk Anda untuk AWS Marketplace](#)
- [Mengirimkan produk Anda untuk dipublikasikan AWS Marketplace](#)

Untuk informasi selengkapnya tentang menyiapkan produk Single-AMI Anda untuk diserahkan AWS Marketplace, lihat sumber daya berikut:

- [Praktik terbaik untuk membangun AMIs untuk digunakan dengan AWS Marketplace](#)
- [Daftar periksa produk AMI untuk AWS Marketplace](#)

- [Persyaratan produk berbasis AMI untuk AWS Marketplace](#)

Tambahkan CloudFormation template ke produk Anda

AWS Marketplace penjual dapat mencantumkan produk berbasis AMI yang dikirim ke AWS Marketplace pembeli dengan menggunakan AWS CloudFormation templat. Menambahkan CloudFormation template ke produk berbasis AMI Anda memungkinkan pembeli Anda untuk menerapkan solusi Anda tanpa harus mengonfigurasi sumber daya dan dependensi secara manual. Anda dapat menggunakan templat untuk menentukan cluster atau arsitektur didistribusikan untuk produk atau untuk memilih kombinasi AMI yang berbeda atau konfigurasi produk. Solusi AMI tunggal dapat berisi maksimal tiga CloudFormation templat.

CloudFormation Template dapat dikonfigurasi untuk mengirimkan satu Amazon Machine Image (AMI) dengan file konfigurasi terkait dan fungsi Lambda. Selain itu, Anda harus menyertakan diagram arsitektur untuk setiap templat.

Topik

- [Menyiapkan CloudFormation templat](#)
- [Diagram arsitektur](#)
- [Mengkonversi CloudFormation template dari produk yang ada](#)
- [Menambahkan komponen aplikasi tanpa server](#)

Menyiapkan CloudFormation templat

Untuk membangun CloudFormation template Anda, Anda harus memenuhi prasyarat template dan memberikan input dan parameter keamanan yang diperlukan. Saat mengirimkan CloudFormation template Anda, gunakan pedoman di bagian berikut.

Prasyarat templat

- Verifikasi bahwa template berhasil diluncurkan melalui AWS CloudFormation konsol di semua Wilayah AWS diaktifkan untuk produk Anda. Anda dapat menggunakan [TaskCat alat ini](#) untuk menguji template Anda.
- AMIs yang disertakan dalam CloudFormation template Anda harus berupa AMI dari produk yang Anda terbitkan atau AMI yang dikelola AWS seperti Amazon Linux 2 terbaru. Jangan sertakan komunitas AMI atau AMI yang dimiliki dan dibagikan oleh Anda atau pihak ketiga lainnya. Untuk

menggunakan AMI yang dikelola AWS, gunakan [parameter publik di AWS Systems Manager Parameter Store](#) alih-alih melakukan hardcoding AMI. IDs Misalnya, dalam CloudFormation template tempat Anda menentukan ID AMI, Anda menggunakan referensi dinamisImageId: `'{{resolve:ssm:/aws/service/ecs/optimized-ami/amazon-linux-2/recommended/image_id}}'`.

- Buat templat sehingga tidak bergantung pada penggunaan di Availability Zone (AZ) tertentu. Tidak semua pelanggan memiliki akses ke semua AZs, dan AZs dipetakan secara berbeda untuk akun yang berbeda.
- Jika Anda membuat solusi kluster menggunakan grup Auto Scaling, sebaiknya akun peristiwa penskalaan. Node baru harus bergabung dengan cluster berjalan secara otomatis.
- Bahkan untuk produk single-node, kami merekomendasikan untuk menggunakan [Grup Auto Scaling](#).
- Jika solusi Anda melibatkan sekelompok peristiwa, pertimbangkan untuk menggunakan grup penempatan jika Anda ingin latensi jaringan rendah, throughput jaringan tinggi, atau keduanya di antara instans.
- Untuk kemudahan peninjauan oleh AWS Marketplace tim dan transparansi kepada pelanggan, kami sarankan Anda menambahkan komentar di UserData bagian Anda.

Persyaratan untuk detail AMI

Note

Jika Anda membuat Amazon Machine Image (AMI) atau AMI dengan CloudFormation halaman [produk server](#) portal penjual dan diminta untuk mengunduh [formulir pemuatan produk](#), lihat [the section called “Persyaratan untuk detail AMI menggunakan formulir pemuatan produk”](#) sebagai gantinya.

[Saat menentukan ImageId properti sumber daya yang menerapkan AMI Anda ke EC2 instans seperti AWS::EC2::Instance, dan AWS::EC2::LaunchTemplate sumber daya AWS::AutoScaling::LaunchConfiguration, Anda harus mereferensikan parameter templat.](#) Jenis parameter harus berupa `aAWS::EC2::Image::Id,AWS::SSM::Parameter::Value<AWS::EC2::Image::Id>`, atau `String`.

Anda dapat memberi nama parameter template ini nama parameter yang valid. AWS Marketplace [menyalin template Anda ke bucket Amazon S3 miliknya sendiri dan mengganti parameter yang ditentukan dengan parameter Parameter Store.AWS Systems Manager](#) AWS Marketplace juga memperbarui deskripsi dan teks kendala untuk membuat nilai yang benar jelas bagi pembeli yang menerapkan template. Saat pembeli menerapkan template Anda, parameter tersebut akan menyelesaikan ID Wilayah AWS AMI -spesifik dari produk yang Anda publikasikan.

[Contoh template berikut menggambarkan parameter template referensi ImageId properti menggunakan fungsi intrinsik Ref.](#)

Contoh YAMG:

```
Parameters:
  ImageId:
    Type: AWS::EC2::Image::Id
    Default: ami-example1234567890
Resources:
  MyInstance:
    Type: AWS::EC2::Instance
    Properties:
      ImageId: !Ref ImageId
```

Contoh JSON:

```
{
  "Parameters": {
    "ImageId": {
      "Type": "AWS::EC2::Image::Id",
      "Default": "ami-example1234567890"
    }
  },
  "Resources": {
    "MyInstance": {
      "Type": "AWS::EC2::Instance",
      "Properties": {
        "ImageId": {
          "Ref": "ImageId"
        }
      }
    }
  }
}
```

Jika Anda menerapkan EC2 instance di dalam [tumpukan bersarang alih-alih tumpukan root](#), ID AMI harus mewarisi nilainya secara dinamis dari tumpukan root. Edit root dan tumpukan bersarang Anda sehingga di tumpukan root, pengaturan nilai parameter template Anda akan menggantikan ID AMI yang digunakan dalam tumpukan bersarang ini.

Persyaratan untuk detail AMI menggunakan formulir pemuatan produk

 Note

Saat Anda membuat Amazon Machine Image (AMI) atau AMI dengan CloudFormation halaman [produk server](#) portal penjual dan tidak segera diminta untuk mengunduh [formulir pemuatan produk](#), lihat [the section called “Persyaratan untuk detail AMI”](#) sebagai gantinya.

AMIs harus dalam tabel pemetaan untuk setiap Wilayah. AWS Marketplace Tim memperbarui AMI IDs setelah mereka dikloning. Sumber AMI Anda harus berada di us-east-1 Wilayah. Wilayah lain dapat menggunakan placeholder.

Contoh YAMG:

```
Mappings:
  RegionMap:
    us-east-1:
      ImageId: ami-0123456789abcdef0
    us-west-1:
      ImageId: ami-xxxxxxxxxxxxxxxxxxxx
    eu-west-1:
      ImageId: ami-xxxxxxxxxxxxxxxxxxxx
    ap-southeast-1:
      ImageId: ami-xxxxxxxxxxxxxxxxxxxx
Resources:
  EC2Instance:
    Type: AWS::EC2::Instance
    Properties:
      ImageId: !FindInMap
        - RegionMap
        - !Ref AWS::Region
        - ImageId
```

Persyaratan untuk templat tumpukan bersarang

Note

Bagian ini hanya berlaku untuk model penetapan harga yang tidak menggunakan [formulir pemuatan produk](#). Untuk model penetapan harga yang menggunakan formulir pemuatan produk, hanya string tetap yang diizinkan untuk `TemplateURL` properti tumpukan bersarang.

Jika template Anda menyertakan [tumpukan bersarang](#), `TemplateURL` properti sumber daya tumpukan bersarang harus mereferensikan parameter template untuk nama bucket Amazon S3, Wilayah bucket, dan awalan key objek Amazon S3. Nama parameter untuk nama bucket harus `MPS3BucketName`, Bucket Region harus `MPS3BucketRegion`, dan untuk object key prefix harus `MPS3KeyPrefix`

Tetapkan nilai default untuk parameter ini agar sesuai dengan bucket Amazon S3 tempat templat bersarang disimpan. Semua template bersarang harus dapat diakses publik. Saat Anda mengirimkan templat untuk diterbitkan, salin templat Anda ke bucket Amazon S3 miliknya sendiri dan memodifikasi properti ketiga parameter tersebut agar nilai default dan nilai yang diizinkan disetel agar sesuai dengan tempat AWS Marketplace salinan disimpan. AWS Marketplace juga memperbarui deskripsi dan teks kendala untuk membuat nilai yang benar jelas bagi pembeli yang menerapkan template.

Jika Anda memiliki beberapa level tumpukan bersarang, semua tumpukan bersarang yang membuat tumpukan bersarang tambahan harus dikonfigurasi sehingga `TemplateURL` properti mewarisi nilai nama bucket Amazon S3, Wilayah bucket Amazon S3, dan kunci objek Amazon S3 secara dinamis dari tumpukan root. Edit root dan tumpukan bersarang Anda sehingga di tumpukan root, atur nilai parameter template `MPS3BucketName`, `MPS3BucketRegion`, dan `MPS3KeyPrefix` ganti nilainya masing-masing di URL yang digunakan dalam tumpukan bersarang ini untuk membuat tumpukan bersarang tambahan.

[Contoh template berikut menggambarkan parameter template referensi `TemplateURL` properti menggunakan fungsi intrinsik `Fn::Sub`.](#)

Contoh YANG:

```
AWSTemplateFormatVersion: '2010-09-09'  
Metadata:  
  AWS::CloudFormation::Interface:  
    ParameterGroups:
```

```

- Label:
  default: AWS Marketplace Parameters
Parameters:
  - ImageId
  - MPS3BucketName
  - MPS3BucketRegion
  - MPS3KeyPrefix
Parameters:
ImageId:
  Type: AWS::EC2::Image::Id
  Default: ami-example1234567890
  Description: The AMI that will be used to launch EC2 resources.
MPS3BucketName:
  Type: String
  Default: sellerbucket
  Description: Name of the S3 bucket for your copy of the nested templates.
MPS3BucketRegion:
  Type: String
  Default: us-east-1
  Description: AWS Region where the S3 bucket for your copy of the nested templates
is hosted.
MPS3KeyPrefix:
  Type: String
  Default: sellerproductfolder/
  Description: S3 key prefix that is used to simulate a folder for your copy of the
nested templates.
Resources:
EC2Instance:
  Type: AWS::EC2::Instance
  Properties:
    ImageId: !Ref ImageId
NestedStack:
  Type: AWS::CloudFormation::Stack
  Properties:
    TemplateURL: !Sub https://${MPS3BucketName}.s3.${MPS3BucketRegion}.
${AWS::URLSuffix}/${MPS3KeyPrefix}nested-template.yaml

```

Contoh JSON:

```

{
  "AWSTemplateFormatVersion": "2010-09-09",
  "Metadata": {
    "AWS::CloudFormation::Interface": {

```

```

    "ParameterGroups": [
      {
        "Label": {
          "default": "AWS Marketplace Parameters"
        },
        "Parameters": [
          "ImageId",
          "MPS3BucketName",
          "MPS3BucketRegion",
          "MPS3KeyPrefix"
        ]
      }
    ]
  },
  "Parameters": {
    "ImageId": {
      "Type": "AWS::EC2::Image::Id",
      "Default": "ami-example1234567890",
      "Description": "The AMI that will be used to launch EC2 resources."
    },
    "MPS3BucketName": {
      "Type": "String",
      "Default": "sellerbucket",
      "Description": "Name of the S3 bucket for your copy of the nested
templates."
    },
    "MPS3BucketRegion": {
      "Type": "String",
      "Default": "us-east-1",
      "Description": "AWS Region where the S3 bucket for your copy of the nested
templates is hosted."
    },
    "MPS3KeyPrefix": {
      "Type": "String",
      "Default": "sellerproductfolder/",
      "Description": "S3 key prefix that is used to simulate a folder for your
copy of the nested templates."
    }
  },
  "Resources": {
    "EC2Instance": {
      "Type": "AWS::EC2::Instance",
      "Properties": {

```

```

        "ImageId": {
            "Ref": "ImageId"
        }
    },
    "NestedStack": {
        "Type": "AWS::CloudFormation::Stack",
        "Properties": {
            "TemplateURL": {
                "Fn::Sub": "https://${MPS3BucketName}.s3.${MPS3BucketRegion}.
${AWS::URLSuffix}/${MPS3KeyPrefix}nested-template.yaml"
            }
        }
    }
}
}

```

Note

[AWS::CloudFormation::Interface](#) digunakan untuk menentukan bagaimana parameter dikelompokkan dan diurutkan di AWS CloudFormation konsol saat pembeli menerapkan template Anda.

Parameter input templat

- Parameter input ke template tidak boleh menyertakan AWS kredensial AWS Marketplace pelanggan (seperti kata sandi, kunci publik, kunci pribadi, atau sertifikat).
- Untuk parameter input sensitif seperti kata sandi, pilih NoEcho properti dan aktifkan ekspresi reguler yang lebih kuat. Untuk parameter input lainnya, atur input yang paling umum bersama dengan teks pembantu yang sesuai.
- Gunakan tipe AWS CloudFormation parameter untuk input jika tersedia.
- Gunakan `AWS::CloudFormation::Interface` untuk kelompok dan mengurutkan parameter masukan.
- Jangan menetapkan nilai default untuk parameter input berikut:

Note

Pelanggan harus menyediakan ini sebagai parameter input.

- Rentang CIDR default yang memungkinkan masuknya ke port akses jarak jauh dari internet publik
- Rentang CIDR default yang memungkinkan masuknya ke port koneksi database dari internet publik
- Kata sandi default untuk pengguna atau database

Jaringan dan parameter keamanan

- Pastikan bahwa port SSH default (22) atau RDP port (3389) tidak terbuka untuk 0.0.0.0.
- Alih-alih menggunakan virtual private cloud (VPC) default, sebaiknya Anda membuat VPC dengan daftar kontrol akses (ACLs) dan grup keamanan yang sesuai.
- Template Anda tidak dapat meminta kunci akses jangka panjang dari pengguna atau membuatnya untuk mengakses AWS sumber daya. Jika aplikasi AMI Anda memerlukan akses ke AWS layanan di akun pembeli, aplikasi tersebut harus menggunakan [peran IAM untuk Amazon EC2](#).
- Tetapkan peran dan kebijakan IAM ke [Berikan hak istimewa yang paling rendah](#) dan aktifkan akses tulis hanya jika benar-benar diperlukan. Misalnya, jika aplikasi Anda hanya perlu S3 : GET, PUT, dan DELETE operasi, tentukan tindakan tersebut saja. Kami tidak merekomendasikan penggunaan S3 : * dalam kasus ini.

Setelah template Anda diterima, AWS Marketplace validasi konfigurasi dan informasi produk dan memberikan umpan balik untuk setiap revisi yang diperlukan.

Diagram arsitektur

Anda harus memberikan diagram arsitektur untuk setiap template. Untuk mempelajari lebih lanjut tentang diagram, lihat [Apa itu diagram arsitektur?](#)

Diagram harus memenuhi kriteria berikut:

- Ilustrasikan penerapan standar pada. AWS

- Menggambarkan secara logis di mana sumber daya digunakan. Misalnya, sumber daya seperti EC2 instans Amazon berada di subnet yang benar.
- Gunakan ikon AWS produk terbaru untuk setiap yang Layanan AWS digunakan melalui AWS CloudFormation template. Untuk mengunduh rangkaian ikon arsitektur saat ini, lihat [Ikon AWS Arsitektur](#).
- Sertakan metadata untuk semua layanan yang digunakan oleh template. AWS CloudFormation
- Sertakan semua jaringan, VPCs, dan subnet yang digunakan oleh template. AWS CloudFormation
- Tampilkan poin integrasi, termasuk aset pihak ketiga, APIs dan aset hibrida lokal.
- Diagram harus berukuran 1100 x 700 piksel. Pertahankan proporsi diagram asli tanpa peregangan atau pemotongan.

Mengkonversi CloudFormation template dari produk yang ada

Note

Bagian ini untuk penjual dengan AMI yang sudah ada dengan CloudFormation produk yang menggunakan [formulir pemuatan produk](#) untuk mempublikasikan templat mereka dan sekarang ingin memperbarui templat itu tanpa menggunakan formulir pemuatan produk. Jika Anda menerbitkan produk baru, lihat [Mempersiapkan CloudFormation templat Anda](#). Jika Anda membuat Amazon Machine Image (AMI) atau AMI dengan CloudFormation halaman [produk server](#) portal penjual dan diminta untuk mengunduh [formulir pemuatan produk](#), lihat [the section called “Persyaratan untuk detail AMI menggunakan formulir pemuatan produk”](#).

Jika Anda ingin menggunakan pengalaman swalayan untuk memperbarui produk yang sudah ada yang sebelumnya menggunakan [formulir pemuatan produk](#) untuk dipublikasikan, Anda harus membuat perubahan pada CloudFormation templat yang ada.

Tabel berikut menjelaskan perbedaan antara menggunakan formulir beban produk dan pengalaman swalayan:

	Formulir beban produk	Pengalaman swalayan
Nilai ImageId properti untuk EC2 sumber daya	Mereferensikan tabel pemetaan untuk ID AMI Anda.	Mereferensikan parameter template untuk ID AMI Anda.

	Formulir beban produk	Pengalaman swalayan
	Untuk informasi selengkapnya, lihat the section called “Persyaratan untuk detail AMI menggunakan formulir pemuatan produk” .	Untuk informasi selengkapnya, lihat the section called “Persyaratan untuk detail AMI” .
Nilai TemplateURL properti untuk tumpukan bersarang	Harus berupa string tetap dan tidak dapat menggunakan fungsi intrinsik.	Bisa dinamis dengan menggunakan fungsi intrinsik. Harus referensi satu set parameter template. Untuk informasi selengkapnya, lihat the section called “Persyaratan untuk templat tumpukan bersarang”

Contoh templat berikut mengilustrasikan contoh produk yang ada yang menggunakan formulir pemuatan produk untuk mempublikasikan template. Dalam contoh ini, ID AMI adalah `ami-example123456` dan template bersarang ada di bucket S3 penjual di lokasi `https://sellerbucket.s3.us-east-1.amazonaws.com/sellerproductfolder/nested-template.yaml`

Contoh YANGM diterbitkan dengan formulir beban produk:

```
AWSTemplateFormatVersion: '2010-09-09'
Mappings:
  RegionMap:
    us-east-1:
      AMI: ami-example123456
Resources:
  EC2Instance:
    Type: AWS::EC2::Instance
    Properties:
      ImageId: !FindInMap
        - RegionMap
        - !Ref AWS::Region
        - AMI
  NestedStack:
```

```
Type: AWS::CloudFormation::Stack
Properties:
  TemplateURL: https://sellerbucket.s3.us-east-1.amazonaws.com/sellerproductfolder/nested-template.yaml
```

Contoh JSON diterbitkan dengan formulir beban produk:

```
{
  "AWSTemplateFormatVersion": "2010-09-09",
  "Mappings": {
    "RegionMap": {
      "us-east-1": {
        "AMI": "ami-example123456"
      }
    }
  },
  "Resources": {
    "EC2Instance": {
      "Type": "AWS::EC2::Instance",
      "Properties": {
        "ImageId": {
          "Fn::FindInMap": [
            "RegionMap",
            {
              "Ref": "AWS::Region"
            }
          ],
          "AMI"
        }
      }
    },
    "NestedStack": {
      "Type": "AWS::CloudFormation::Stack",
      "Properties": {
        "TemplateURL": "https://sellerbucket.s3.us-east-1.amazonaws.com/sellerproductfolder/nested-template.yaml"
      }
    }
  }
}
```

Contoh template berikut menggambarkan perubahan yang diperlukan untuk menggunakan pengalaman swalayan untuk memperbarui produk.

Contoh YAMG diterbitkan dengan pengalaman swalayan:

```
AWSTemplateFormatVersion: '2010-09-09'
Metadata:
  AWS::CloudFormation::Interface:
    ParameterGroups:
      - Label:
          default: AWS Marketplace Parameters
        Parameters:
          - ImageId
          - MPS3BucketName
          - MPS3BucketRegion
          - MPS3KeyPrefix
Parameters:
  ImageId:
    Type: AWS::EC2::Image::Id
    Default: ami-example123456
    Description: The AMI that will be used to launch EC2 resources.
  MPS3BucketName:
    Type: String
    Default: sellerbucket
    Description: Name of the S3 bucket for your copy of the nested templates.
  MPS3BucketRegion:
    Type: String
    Default: us-east-1
    Description: AWS Region where the S3 bucket for your copy of the nested templates
is hosted.
  MPS3KeyPrefix:
    Type: String
    Default: sellerproductfolder/
    Description: S3 key prefix that is used to simulate a folder for your copy of the
nested templates.
Resources:
  EC2Instance:
    Type: AWS::EC2::Instance
    Properties:
      ImageId: !Ref ImageId
  NestedStack:
    Type: AWS::CloudFormation::Stack
    Properties:
      TemplateURL: !Sub https://{MPS3BucketName}.s3.{MPS3BucketRegion}.
${AWS::URLSuffix}/{MPS3KeyPrefix}nested-template.yaml
```

Contoh JSON diterbitkan dengan pengalaman swalayan:

```
{
  "AWSTemplateFormatVersion": "2010-09-09",
  "Metadata": {
    "AWS::CloudFormation::Interface": {
      "ParameterGroups": [
        {
          "Label": {
            "default": "AWS Marketplace Parameters"
          },
          "Parameters": [
            "ImageId",
            "MPS3BucketName",
            "MPS3BucketRegion",
            "MPS3KeyPrefix"
          ]
        }
      ]
    }
  },
  "Parameters": {
    "ImageId": {
      "Type": "AWS::EC2::Image::Id",
      "Default": "ami-example123456",
      "Description": "The AMI that will be used to launch EC2 resources."
    },
    "MPS3BucketName": {
      "Type": "String",
      "Default": "sellerbucket",
      "Description": "Name of the S3 bucket for your copy of the nested
templates."
    },
    "MPS3BucketRegion": {
      "Type": "String",
      "Default": "us-east-1",
      "Description": "AWS Region where the S3 bucket for your copy of the nested
templates is hosted."
    },
    "MPS3KeyPrefix": {
      "Type": "String",
      "Default": "sellerproductfolder/",
      "Description": "S3 key prefix that is used to simulate a folder for your
copy of the nested templates."
    }
  }
}
```

```

    }
  },
  "Resources": {
    "EC2Instance": {
      "Type": "AWS::EC2::Instance",
      "Properties": {
        "ImageId": {
          "Ref": "ImageId"
        }
      }
    }
  },
  "NestedStack": {
    "Type": "AWS::CloudFormation::Stack",
    "Properties": {
      "TemplateURL": {
        "Fn::Sub": "https://${MPS3BucketName}.s3.${MPS3BucketRegion}.
${AWS::URLSuffix}/${MPS3KeyPrefix}nested-template.yaml"
      }
    }
  }
}

```

Menambahkan komponen aplikasi tanpa server

Important

AWS Marketplace tidak lagi mendukung penerbitan produk baru dengan CloudFormation template yang menyebarkan sumber daya dari AWS Serverless Application Repository. Penjual dapat terus mempublikasikan produk yang ada dengan CloudFormation templat yang menyebarkan sumber daya dari AWS Serverless Application Repository hingga tanggal mendatang untuk diumumkan.

Anda dapat membuat produk yang menyertakan Amazon Machine Image (AMI), dikirimkan menggunakan AWS CloudFormation template, dengan komponen tanpa server yang dimasukkan ke dalam produk. Misalnya, Anda dapat membuat produk dengan AMI yang dikonfigurasi sebagai server pengontrol dan mengirimkannya sebagai AWS CloudFormation tumpukan. AWS CloudFormation Template yang digunakan untuk membuat tumpukan dapat menyertakan definisi untuk mengatur AWS Lambda fungsi yang dipicu oleh peristiwa di server. Bila Anda menggunakan pendekatan

ini untuk merancang produk Anda, Anda dapat menyederhanakan arsitektur dan memudahkan pembeli Anda untuk memulai. Pendekatan ini juga dapat membuat lebih mudah bagi Anda untuk memperbarui produk Anda. Bagian berikut ini menunjukkan kepada Anda cara membuat dan menawarkan jenis produk ini.

Untuk informasi tentang membuat AMI untuk produk Anda, lihat [Produk berbasis AMI di AWS Marketplace](#). Untuk informasi tentang melengkapi AWS CloudFormation template untuk produk Anda, lihat [Tambahkan CloudFormation template ke produk Anda](#).

Ketika Anda mendefinisikan aplikasi tanpa server Anda, Anda menggunakan AWS Serverless Application Model (AWS SAM) template yang Anda simpan di. AWS Serverless Application Repository AWS SAM adalah kerangka kerja sumber terbuka untuk membangun aplikasi tanpa server. Selama penyebaran, AWS SAM mengubah dan memperluas sintaks menjadi AWS Serverless Application Model sintaks. AWS CloudFormation Parameter AWS Serverless Application Repository adalah repositori dikelola untuk aplikasi tanpa server. Hal ini memungkinkan Anda untuk menyimpan dan berbagi aplikasi yang dapat digunakan kembali sehingga pembeli dapat merakit dan men-deploy arsitektur tanpa server.

Note

- AWS Marketplace meninjau dan memvalidasi produk Anda sebelum listing Anda dibuat. Jika ada masalah yang harus Anda selesaikan sebelum penawaran terdaftar, kami akan mengirimkan pesan email kepada Anda.
- Sebagai bagian dari memenuhi langganan, kami menyalin, aplikasi tanpa server AMIs, dan AWS CloudFormation templat ke repositori yang AWS Marketplace dimiliki di masing-masing. Wilayah AWS Saat pembeli berlangganan produk Anda, kami memberi mereka akses, dan juga mbucketitahu mereka saat Anda memperbarui perangkat lunak Anda.

Topik

- [Langkah 1: Buat aplikasi tanpa server](#)
- [Langkah 2: Publikasikan aplikasi Anda ke repositori](#)
- [Langkah 3: Buat CloudFormation template](#)
- [Langkah 4: Kirim file CloudFormation template dan konfigurasi Anda](#)
- [Langkah 5: Perbarui izin AWS Serverless Application Repository aplikasi Anda](#)
- [Langkah 6: Bagikan AMI Anda](#)

- [Langkah 7: Kirim CloudFormation produk Anda dengan AMI dan aplikasi tanpa server](#)

Langkah 1: Buat aplikasi tanpa server

Langkah pertama Anda adalah mengemas AWS Lambda fungsi yang digunakan untuk membuat aplikasi tanpa server Anda. Aplikasi Lambda adalah kombinasi dari fungsi Lambda, sumber peristiwa, dan sumber daya lainnya yang bekerja sama untuk melakukan tugas. Aplikasi tanpa server dapat sesederhana satu fungsi Lambda, atau dapat berisi beberapa fungsi dengan sumber daya lain, seperti, database APIs, dan pemetaan sumber peristiwa.

Gunakan AWS SAM untuk menentukan model untuk aplikasi tanpa server Anda. Untuk deskripsi nama dan jenis properti, lihat [AWS::Serverless::Application](#) di AWS Labs pada GitHub. Berikut ini adalah contoh AWS SAM template dengan fungsi Lambda tunggal dan peran AWS Identity and Access Management (IAM).

```
AWSTemplateFormatVersion: '2010-09-09'
Transform: AWS::Serverless-2016-10-31
Description: An example of SAM template with Lambda function and IAM role

Resources:
  SampleFunction:
    Type: AWS::Serverless::Function
    Properties:
      Handler: 'com.sampleproject.SampleHandler::handleRequest'
      Runtime: java8
      CodeUri: 's3://amzn-s3-demo-bucket/2EXAMPLE-1234-4b12-ac37-515EXAMPLEe5-
Lambda.zip'
      Description: Sample Lambda function
      Timeout: 120
      MemorySize: 1024
      Role:
        Fn::GetAtt: [SampleFunctionRole, Arn]

# Role to execute the Lambda function
SampleFunctionRole:
  Type: "AWS::IAM::Role"
  Properties:
    AssumeRolePolicyDocument:
      Statement:
        - Effect: "Allow"
          Principal:
```

```
Service:
  - "lambda.amazonaws.com"
Action: "sts:AssumeRole"
ManagedPolicyArns:
  - "arn:aws:iam::aws:policy/service-role/AWSLambdaBasicExecutionRole"
Policies:
  - PolicyName: SFNXDeployWorkflowDefinitionPolicy
    PolicyDocument:
      Statement:
        - Effect: "Allow"
          Action:
            - "s3:Get*"
          Resource: "*"
RoleName: "SampleFunctionRole"
```

Langkah 2: Publikasikan aplikasi Anda ke repositori

Untuk menerbitkan aplikasi, Anda memuat naik kod aplikasi terlebih dahulu. Simpan artefak kode Anda (misalnya, fungsi Lambda, skrip, file konfigurasi) di bucket Amazon S3 yang dimiliki akun Anda. Ketika Anda mengunggah aplikasi Anda, awalnya disetel ke pribadi, yang berarti bahwa itu hanya tersedia untuk Akun AWS yang membuatnya. Anda harus membuat kebijakan IAM yang memberikan AWS Serverless Application Repository izin untuk mengakses artefak yang Anda unggah.

Untuk mempublikasikan aplikasi tanpa server Anda ke repositori aplikasi tanpa server

1. Buka konsol Amazon S3 di <https://console.aws.amazon.com/s3/>
2. Pilih bucket Amazon S3 yang Anda gunakan untuk mengemas aplikasi Anda.
3. Pilih tab Izin.
4. Pilih Kebijakan Bucket.
5. Salin dan tempel contoh pernyataan kebijakan berikut.

Note

Pernyataan kebijakan contoh akan menghasilkan kesalahan sampai nilai untuk `aws:SourceAccount` dan `Resource` diperbarui dalam langkah-langkah berikut.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "serverlessrepo.amazonaws.com"
      },
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::amzn-s3-demo-bucket/*",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "123456789012"
        }
      }
    }
  ]
}
```

- a. Ganti amzn-s3-demo-bucket di nilai Resource properti dengan nama bucket untuk bucket Anda.
 - b. Ganti **123456789012** Condition elemen dengan Akun AWS ID Anda. ConditionElemen memastikan bahwa AWS Serverless Application Repository hanya memiliki izin untuk mengakses aplikasi dari yang ditentukan Akun AWS.
6. Pilih Simpan.
 7. Buka AWS Serverless Application Repository konsol di <https://console.aws.amazon.com/serverlessrepo>.
 8. Pada halaman Aplikasi Saya, pilih Publikasikan aplikasi.
 9. Lengkapi bidang yang wajib diisi dan bidang opsional, yang sesuai. Bidang yang wajib diisi:
 - Nama aplikasi
 - Pengarang
 - Deskripsi

- URL kode sumber
- Templat SAM

10. PilihPublikasikan Aplikasi.

Untuk mempublikasikan versi berikutnya dari aplikasi Anda

1. Buka AWS Serverless Application Repository konsol di <https://console.aws.amazon.com/serverlessrepo>.
2. Di panel navigasi, dariAplikasi saya, pilih aplikasi.
3. PilihPublikasikan versi baru.

Untuk informasi selengkapnya, lihat [Menerbitkan Aplikasi tanpa server Menggunakan CLI AWS SAM](#).

Langkah 3: Buat CloudFormation template

Untuk membangun CloudFormation template Anda, Anda harus memenuhi prasyarat template dan memberikan input dan parameter keamanan yang diperlukan. Untuk informasi selengkapnya, lihat [Anatomi templat](#) di Panduan AWS CloudFormation Pengguna.

Dalam CloudFormation template Anda, Anda dapat mereferensikan aplikasi tanpa server dan AMI Anda. Anda juga dapat menggunakan CloudFormation templat bersarang dan referensi aplikasi tanpa server baik di template root maupun templat bersarang. Untuk mereferensikan aplikasi tanpa server, Anda menggunakan template. AWS SAM Anda dapat secara otomatis membuat AWS SAM template untuk aplikasi Anda dari AWS Serverless Application Repository. Berikut adalah contoh.

```
AWSTemplateFormatVersion: '2010-09-09'
Transform: AWS::Serverless-2016-10-31
Description: An example root template for a SAR application

Resources:
  SampleSARApplication:
    Type: AWS::Serverless::Application
    Properties:
      Location:
        ApplicationId: arn:aws:serverlessrepo:us-east-1:1234567890:applications/
TestApplication
  SemanticVersion: 1.0.0
  SampleEC2Instance:
    Type: AWS::EC2::Instance
```

```
Properties:
  ImageId: "ami-79fd7eee"
  KeyName: "testkey"
  BlockDeviceMappings:
    - DeviceName: "/dev/sdm"
      Ebs:
        VolumeType: "io1"
        Iops: "200"
        DeleteOnTermination: "false"
        VolumeSize: "20"
    - DeviceName: "/dev/sdk"
      NoDevice: {}
```

AWS SAM Template berisi elemen-elemen berikut:

- **ApplicationID**— Nama Sumber Daya Amazon (ARN) aplikasi Anda. Informasi ini terletak di Aplikasi saya Bagian dari AWS Serverless Application Repository.
- **SemanticVersion**— Versi aplikasi tanpa server Anda. Anda dapat menemukan ini dari Aplikasi saya Bagian dari AWS Serverless Application Repository.
- **Parameter**(opsional) — Parameter aplikasi.

Note

Untuk **ApplicationID** dan **SemanticVersion**, [fungsi intrinsik](#) tidak didukung. Anda harus hardcode string itu. **ApplicationID** ini diperbarui saat dikloning oleh AWS Marketplace.

Jika Anda berencana untuk referensi konfigurasi dan file skrip dalam CloudFormation template Anda, gunakan format berikut. Untuk template bersarang (`AWS::CloudFormation::Stack`), hanya `TemplateURLs` tanpa fungsi intrinsik yang didukung. Catatan `Parameters` konten dalam templat.

```
AWSTemplateFormatVersion: '2010-09-09'
Metadata:
  Name: Seller test product
Parameters:
  CFRefFilesBucket:
    Type: String
    Default: "seller-bucket"
  CFRefFilesBucketKeyPrefix:
    Type: String
```

```

    Default: "cftsolutionFolder/additionCFfiles"
Resources:
  TestEc2:
    Type: AWS::EC2::Instance
    Metadata:
      AWS::CloudFormation::Init:
        addCloudAccount:
          files:
            /etc/cfn/set-aia-settings.sh:
              source:
                Fn::Sub:
                  - https://${CFTRefFilesBucket}.${S3Region}amazonaws.com/
                    ${CFTRefFilesBucketKeyPrefix}/sampleScript.sh
                  - S3Region:
                      !If
                        - GovCloudCondition
                        - s3-us-gov-west-1
                        - s3
                  owner: root
                  mode: '000700'
                  authentication: Amazon S3AccessCreds
          ..
          ..
          ..
  SampleNestedStack:
    Type: AWS::CloudFormation::Stack
    Properties:
      TemplateURL: 'https://sellerbucket.s3.amazon.com/sellerproductfolder/
nestedCft.template'
      Parameters:
        SampleParameter: 'test'
Transform: AWS::Serverless-2016-10-31

```

Langkah 4: Kirim file CloudFormation template dan konfigurasi Anda

Untuk mengirimkan file CloudFormation template, konfigurasi, dan skrip, berikan AWS Marketplace izin untuk membaca bucket Amazon S3 tempat file ini disimpan. Untuk melakukannya, memperbarui kebijakan bucket Anda untuk menyertakan izin berikut.

JSON

```
{
```

```

"Version": "2012-10-17",
"Statement": [
  {
    "Effect": "Allow",
    "Principal": {
      "Service": "assets.marketplace.amazonaws.com"
    },
    "Action": ["s3:GetObject", "s3:ListBucket"],
    "Resource": ["arn:aws:s3:::amzn-s3-demo-bucket",
      "arn:aws:s3:::amzn-s3-demo-bucket/*"]
  }
]
}

```

Langkah 5: Perbarui izin AWS Serverless Application Repository aplikasi Anda

Untuk mengirimkan AWS Serverless Application Repository aplikasi Anda AWS Marketplace, Anda harus memberikan AWS Marketplace izin untuk membaca aplikasi Anda. Untuk melakukannya, tambahkan izin ke kebijakan yang terkait dengan aplikasi tanpa server Anda. Ada dua cara untuk memperbarui kebijakan aplikasi Anda:

- AWS Serverless Application Repository Pilih aplikasi tanpa server Anda dari daftar. Pilih Pembagian tab, dan pilih Buat Pernyataan. Pada halaman konfigurasi Pernyataan, masukkan prinsip layanan berikut **assets.marketplace.amazonaws.com**, di bidang ID Akun. Lalu, pilih Simpan.
- Gunakan AWS CLI perintah berikut untuk memperbarui kebijakan aplikasi Anda.

```

aws serverlessrepo put-application-policy \
--region region \
--application-id application-arn \
--statements Principals=assets.marketplace.amazonaws.com,Actions=Deploy

```

Langkah 6: Bagikan AMI Anda

Semua yang AMIs dibangun dan diserahkan AWS Marketplace harus mematuhi semua kebijakan produk. Pemindaian AMI swalayan tersedia di Portal Manajemen AWS Marketplace Dengan fitur ini, Anda dapat memulai pemindaian Anda. AMIs Anda menerima hasil pemindaian dengan cepat (biasanya, dalam waktu kurang dari satu jam) dengan umpan balik yang jelas di satu lokasi. Setelah

AMI Anda berhasil dipindai, kirimkan AMI untuk diproses oleh tim Operasi AWS Marketplace Penjual dengan mengunggah formulir pemuatan produk Anda.

Langkah 7: Kirim CloudFormation produk Anda dengan AMI dan aplikasi tanpa server

Ingatlah hal-hal berikut sebelum Anda mengirimkan produk Anda:

- Anda harus memberikan diagram arsitektur untuk setiap template. Diagram harus menggunakan ikon AWS produk untuk setiap AWS layanan yang digunakan melalui CloudFormation template. Juga, diagram harus menyertakan metadata untuk layanan. Untuk mengunduh ikon AWS arsitektur resmi kami, lihat [Ikon AWS Arsitektur](#).
- Prakiraan biaya infrastruktur untuk setiap templat yang ditampilkan kepada pembeli didasarkan pada prakiraan yang Anda berikan dengan menggunakan [AWS Kalkulator Harga](#). Dalam estimasi, sertakan daftar layanan yang akan digunakan sebagai bagian dari template, bersama dengan nilai default untuk penerapan tipikal.
- Lengkapi formulir beban produk. Anda dapat menemukan formulir beban produk dari Portal Manajemen AWS Marketplace. Formulir beban produk yang berbeda diperlukan untuk produk AMI tunggal dan beberapa produk AMI. Dalam formulir pemuatan produk, Anda akan memberikan URL publik ke CloudFormation template Anda. CloudFormation template harus diserahkan dalam bentuk URL publik.
- Gunakan Portal Manajemen AWS Marketplace untuk mengirimkan daftar Anda. Dari Aset, pilih Unggahan file, melampirkan file Anda, kemudian pilih Unggah. Setelah kami menerima template dan metadata Anda, AWS mulailah memproses permintaan Anda.

Setelah Anda mengirimkan daftar Anda, AWS Marketplace meninjau dan memvalidasi formulir pemuatan produk. Selain itu, AWS Marketplace regionalisasi AMIs dan aplikasi tanpa server, dan memperbarui pemetaan regional untuk template Anda atas nama Anda AWS CloudFormation. Jika terjadi masalah, tim Operasi AWS Marketplace Penjual akan menghubungi Anda melalui email.

Mengelola produk berbasis AMI sebagai penjual AWS Marketplace

Sebagai AWS Marketplace penjual, Anda dapat mengelola dan memperbarui produk [Amazon Machine Images \(AMIs\)](#) tunggal Anda. Produk berbasis AMI mencakup satu set atau lebih versi perangkat lunak dan metadata tentang produk secara keseluruhan. Saat membuat produk, Anda mengonfigurasi propertinya AWS Marketplace, termasuk nama produk, deskripsi, dan harga Anda. Anda juga menentukan kategori yang sesuai untuk produk Anda dan menambahkan kata kunci

sehingga produk Anda muncul dalam pencarian yang relevan. Setelah Anda membuat, produk Single-AMI Anda, Anda dapat mengirimkan permintaan perubahan untuk membuat perubahan pada produk atau versi Anda.

Tipe perubahan yang dapat Anda minta untuk produk berbasis AMI meliputi:

- Memperbarui informasi produk yang ditampilkan kepada pembeli.
- Memperbarui informasi versi yang ditampilkan kepada pembeli.
- Menambahkan versi baru produk Anda.
- Membatasi versi sehingga pembeli baru tidak dapat lagi mengakses versi tersebut.
- Wilayah AWS Perbarui tempat produk tersedia.
- Memperbarui harga dan contoh tipe untuk produk.
- Hapus produk dari AWS Marketplace.

Topik berikut menunjukkan untuk mengelola dan memperbarui satu produk AMI.

Topik

- [Membuat permintaan perubahan untuk produk berbasis AMI di AWS Marketplace](#)
- [Memperbarui visibilitas produk berbasis AMI](#)
- [Menambahkan dan membatasi instance AMI untuk AWS Marketplace](#)
- [Mengelola versi untuk produk berbasis AMI di AWS Marketplace](#)
- [Memperbarui informasi produk berbasis AMI tentang AWS Marketplace](#)
- [Mengelola ketersediaan produk berbasis AMI menurut Wilayah AWS dan negara](#)
- [Memperbarui perjanjian lisensi pengguna akhir \(EULA\) untuk produk berbasis AWS Marketplace AMI Anda](#)
- [Memperbarui kebijakan pengembalian dana untuk produk berbasis AMI Anda di AWS Marketplace](#)
- [Memberikan AWS Marketplace akses ke AMI Anda](#)
- [Menghapus produk dari AWS Marketplace](#)
- [Memecahkan masalah kesalahan umum untuk permintaan perubahan pada AWS Marketplace](#)

Membuat permintaan perubahan untuk produk berbasis AMI di AWS Marketplace

Untuk membuat perubahan pada produk atau versi AWS Marketplace, Anda mengirimkan permintaan perubahan melalui Portal Manajemen AWS Marketplace. Permintaan perubahan ditambahkan ke antrian dan dapat memakan waktu dari hitungan menit hingga hitungan untuk diselesaikan, tergantung tipe permintaannya. Anda dapat melihat status permintaan di Portal Manajemen AWS Marketplace. Topik ini menyediakan prosedur yang dapat Anda gunakan untuk membuat permintaan perubahan untuk produk Single-AMI AWS Marketplace, termasuk opsi untuk menggunakan pengalaman swalayan.

Anda dapat membuat permintaan perubahan untuk situasi berikut:

- Anda menyimpan langkah-langkah yang sedang berlangsung, tetapi tidak menyelesaikan seluruh proses, saat menggunakan pengalaman layanan mandiri untuk membuat daftar produk Single-AMI. Untuk menyelesaikan langkah-langkah yang tersisa, Anda membuat permintaan perubahan.
- Anda ingin melakukan modifikasi pada informasi produk untuk produk Anda yang berada dalam keadaan Terbatas atau Publik. Untuk memperbarui informasi, Anda membuat permintaan perubahan. Untuk informasi selengkapnya tentang jenis perubahan yang dapat Anda minta untuk produk berbasis AMI, lihat. [Buat permintaan perubahan](#)

Note

Selain itu Portal Manajemen AWS Marketplace, Anda juga dapat membuat permintaan perubahan dengan menggunakan [API AWS Marketplace Katalog](#).

Topik

- [Buat permintaan perubahan dengan menggunakan layanan mandiri](#)
- [Buat permintaan perubahan](#)
- [Dapatkan status permintaan perubahan](#)
- [Sumber daya tambahan](#)

Buat permintaan perubahan dengan menggunakan layanan mandiri

Untuk membuat modifikasi pada versi atau informasi produk, Anda membuat permintaan perubahan di Portal Manajemen AWS Marketplace. Permintaan perubahan adalah blok bangunan dari daftar layanan mandiri yang Anda gunakan untuk membuat perubahan pada produk Anda. Setiap kali Anda memilih Simpan dan keluar dari langkah-langkah atau pilih Kirim untuk pembaruan apa pun, Anda membuat permintaan perubahan. Anda dapat menemukan permintaan Anda di tab Portal Manajemen AWS Marketplace [Permintaan](#).

Untuk membuat permintaan perubahan menggunakan swalayan

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/>, dan masuk ke akun penjual Anda, lalu buka halaman [Produk Server](#).
2. Pada tab Produk Server, pilih produk yang ingin Anda modifikasi.
3. Pilih satu opsi dari daftar tarik turun Permintaan perubahan.
4. Setelah Anda membuat permintaan perubahan, ada waktu tunggu untuk sistem memproses permintaan Anda, tercermin Dalam Tinjauan. Ketika permintaan selesai, itu akan menghasilkan Berhasil atau Gagal.
 - Setelah permintaan diajukan, permintaan mulai diproses melalui status ini: Sedang ditinjau, Mempersiapkan perubahan, dan Menerapkan perubahan.
 - Berhasil berarti bahwa perubahan yang diminta telah diproses dan perubahan tercermin dalam sistem.
 - Gagal berarti ada yang tidak beres dengan permintaan, sehingga perubahan tidak diproses. Jika status Gagal, Anda dapat memilih permintaan untuk menemukan kode kesalahan yang memberikan rekomendasi tentang cara memperbaiki kesalahan. Pada titik ini, Anda dapat memecahkan masalah kesalahan dan membuat permintaan baru untuk perubahan. Untuk membuat proses lebih cepat, Anda dapat memilih Salin ke permintaan baru untuk menyalin rincian permintaan yang gagal. Kemudian, Anda dapat melakukan penyesuaian dan mengirimkan kembali permintaan.

Buat permintaan perubahan

Important

Pada 15 Juni 2023, AWS Marketplace akan menghentikan prosedur berikut. Setelah 15 Juni 2023, gunakan [the section called “Buat permintaan perubahan dengan menggunakan layanan mandiri”](#) prosedur ini.

Untuk membuat modifikasi pada versi atau informasi produk, Anda membuat permintaan perubahan di Portal Manajemen AWS Marketplace.

Untuk membuat permintaan perubahan

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/>, dan masuk ke akun penjual Anda, lalu buka halaman [Produk Server](#).
2. Pada tab Produk Server, pilih produk yang ingin Anda modifikasi.
3. Pilih opsi dari daftar tarik-turun Permintaan perubahan.

Untuk sebagian besar permintaan perubahan, Anda cukup mengisi formulir di antarmuka pengguna dan mengirimkannya. Namun, untuk perubahan tertentu, Anda harus mengunduh, menyelesaikan, kemudian mengunggah formulir beban produk (PLF). Ini adalah spreadsheet yang berisi formulir untuk Anda isi dengan informasi yang diperlukan. Ketika Anda memilih salah satu permintaan perubahan ini, Anda diminta untuk mengunduh PLF yang benar untuk permintaan yang Anda coba buat. PLF telah diisi sebelumnya dengan informasi dari detail produk Anda yang sudah ada. Anda dapat mengunggah PLF Anda yang sudah selesai ke halaman [Unggah Portal Manajemen AWS Marketplace file](#).

Note

Kami sangat menyarankan agar Anda mengunduh dan menggunakan PLF terbaru. Formulir diperbarui secara berkala dengan informasi baru, termasuk jenis instance dan Wilayah AWS saat tersedia. Anda dapat menemukan PLF terbaru untuk produk dari halaman Produk Server, dengan memilih produk, kemudian memilih Unduh Formulir Beban Produk.

Untuk informasi selengkapnya tentang status permintaan perubahan, lihat [the section called “Dapatkan status permintaan perubahan”](#). Untuk mengetahui potensi masalah dengan permintaan perubahan, lihat [Memecahkan masalah kesalahan umum untuk permintaan perubahan pada AWS Marketplace](#).

Dapatkan status permintaan perubahan

Important

Pada 15 Juni 2023, AWS Marketplace akan menghentikan prosedur berikut. Prosedur ini tidak lagi diperlukan untuk pengalaman swalayan.

Setelah Anda mengirimkan permintaan perubahan, Anda dapat melihat status permintaan Anda dari tab Permintaan pada halaman [Produk Server](#) Portal Manajemen AWS Marketplace. Status dapat merupakan hal berikut:

- Dalam tinjauan berarti bahwa permintaan Anda sedang ditinjau. Beberapa permintaan memerlukan peninjauan manual oleh AWS Marketplace tim tetapi sebagian besar ditinjau secara otomatis dalam sistem.
- Berhasil berarti bahwa permintaan Anda sudah lengkap. Produk atau versi Anda telah diperbarui sesuai permintaan.
- Butuh tindakan berarti Anda perlu memperbarui permintaan Anda untuk memperbaiki masalah atau menjawab pertanyaan tentang permintaan tersebut. Pilih permintaan untuk melihat detailnya, termasuk masalah apa pun.
- Gagal berarti ada yang tidak beres dengan permintaan tersebut, dan Anda harus membuat permintaan baru untuk perubahan tersebut, dengan data yang sama.

Sumber daya tambahan

Untuk detail selengkapnya tentang permintaan perubahan untuk jenis pembaruan tertentu, lihat sumber daya berikut:

- [Memperbarui informasi produk berbasis AMI tentang AWS Marketplace](#)
- [Perbarui informasi versi](#)
- [Tambahkan versi baru](#)

- [Batasi versi](#)

Memperbarui visibilitas produk berbasis AMI

Sebagai AWS Marketplace penjual, Anda dapat memperbarui visibilitas produk Anda untuk mengubah pembeli mana yang dapat melihat produk AMI tunggal Anda. AWS Marketplace Jika status visibilitas disetel ke Publik, produk Anda dapat dilihat oleh semua AWS Marketplace pembeli. Jika visibilitas produk Anda diatur ke Limited, produk Anda hanya dapat dilihat oleh daftar Akun AWS IDs yang Anda izinkan. Anda juga dapat mengelola dan memperbarui daftar yang diizinkan ini Akun AWS IDs yang dapat melihat produk Anda. Bagian berikut menunjukkan cara memperbarui visibilitas produk dan daftar izin visibilitas terbatas.

Topik

- [Perbarui visibilitas produk](#)
- [Perbarui daftar yang diizinkan \(akun pratinjau\)](#)

Perbarui visibilitas produk

Untuk mengubah pembeli mana yang dapat melihat produk Anda AWS Marketplace, Anda dapat menggunakan Perbarui visibilitas.

Untuk memperbarui visibilitas

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/>, lalu masuk ke akun penjual Anda.
2. Buka halaman [Produk Server](#), pada tab Produk server saat ini, pilih produk yang ingin Anda modifikasi.
3. Dari menu tarik-turun Permintaan perubahan, pilih Perbarui visibilitas.

Note

Anda dapat meminta agar produk dipindahkan dari status Terbatas ke status Publik dengan menggunakan permintaan perubahan ini. Namun, permintaan perubahan harus melalui proses persetujuan tim Operasi AWS Marketplace Penjual untuk dipindahkan ke Publik.

4. Pilih Kirim permintaan perubahan untuk mengirimkan permintaan Anda untuk ditinjau.

5. Verifikasi bahwa tab Permintaan menampilkan status Permintaan sebagai Dalam peninjauan. Ketika permintaan selesai, status menjadi Berhasil.

Perbarui daftar yang diizinkan (akun pratinjau)

Untuk mengubah daftar Akun AWS IDs yang dapat melihat produk Anda dalam keadaan Terbatas, gunakan Perbarui daftar yang diizinkan.

Untuk memperbarui daftar yang diizinkan

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/>, lalu masuk ke akun penjual Anda.
2. Buka halaman [Produk Server](#), dan pada tab Produk server saat ini, pilih produk yang ingin Anda modifikasi.
3. Dari menu tarik-turun Permintaan perubahan, pilih Perbarui daftar yang diizinkan. Daftar saat ini disediakan dengan daftar akun yang saat ini diizinkan terdaftar.
4. Tambahkan Akun AWS IDs yang lebih disukai untuk visibilitas dan pisahkan IDs dengan koma.
5. Pilih Kirim permintaan perubahan untuk mengirimkan permintaan Anda untuk ditinjau.
6. Verifikasi bahwa tab Permintaan menampilkan status Permintaan sebagai Dalam peninjauan. Ketika permintaan selesai, status menjadi Berhasil.

Menambahkan dan membatasi instance AMI untuk AWS Marketplace

Sebagai AWS Marketplace penjual, Anda dapat mengelola instance mana yang dapat digunakan pembeli untuk produk Amazon Machine Image (AMI) tunggal Anda. Anda dapat menambahkan instance baru untuk produk AMI tunggal Anda yang dapat digunakan pembeli. Demikian pula, jika Anda ingin mencegah pembeli baru menggunakan produk AMI tunggal Anda dari instance tertentu, Anda dapat membatasi instance tersebut.

Untuk informasi selengkapnya tentang jenis EC2 instans Amazon, lihat [Jenis instans yang tersedia](#) di Panduan EC2 Pengguna Amazon.

Bagian berikut menjelaskan cara menambahkan dan membatasi instance.

Topik

- [Menambahkan sebuah instance](#)
- [Membatasi sebuah instance](#)

Menambahkan sebuah instance

Anda dapat menambahkan instance baru yang dapat digunakan pembeli sebagai Single-AMI.

Untuk menambahkan sebuah instance

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/>, lalu masuk ke akun penjual Anda.
2. Buka halaman [Produk Server](#), dan pada tab Produk server saat ini, pilih produk yang ingin Anda modifikasi.
3. Dari menu tarik-turun Permintaan perubahan, pilih Tambahkan instance.
4. Pilih arsitektur instance.
5. Pilih jenis instance yang ingin Anda tambahkan dari daftar instance yang tersedia.
6. Pilih Kirim permintaan untuk mengirimkan permintaan Anda untuk ditinjau.
7. Verifikasi bahwa tab Permintaan menampilkan status Permintaan sebagai Dalam peninjauan. Ketika permintaan selesai, status menjadi Berhasil.

Note

- Jika model harga Anda saat ini tidak gratis atau menggunakan model Bring Your Own License (BYOL), Anda juga harus menambahkan harga.
- Jika Anda membuat instans Tambah dengan harga untuk instans baru atau Perbarui harga untuk menaikkan harga, Anda tidak dapat menggunakan layanan mandiri untuk Menambahkan instans dalam 90 hari mulai dari hari Anda membuat perubahan. Untuk melakukan perubahan ini, hubungi [tim Operasi AWS Marketplace Penjual](#).
- Saat Anda menambahkan dukungan untuk jenis instans baru, pelanggan yang sudah berlangganan penawaran pribadi untuk produk Anda tidak akan dapat meluncurkan instance yang baru ditambahkan secara otomatis. Anda harus membuat penawaran pribadi lain dengan contoh yang Anda ingin pelanggan akses. Setelah menerima penawaran baru, pelanggan dapat meluncurkan instance yang baru ditambahkan. Pelanggan yang berlangganan produk Anda di masa mendatang juga dapat meluncurkan instance, selama instance tersebut termasuk dalam penawaran pribadi. Untuk informasi selengkapnya tentang cara membuat penawaran pribadi baru, lihat [Mengubah perjanjian di AWS Marketplace](#) nanti di panduan ini.

Membatasi sebuah instance

Untuk mencegah pembeli baru menggunakan instance produk AMI, Anda dapat membatasi instance. Anda dapat menambahkan instance kembali di lain waktu, jika diperlukan. Pengguna AMI tunggal yang ada pada instans terbatas dapat terus menggunakan produk dari Wilayah selama masa langganan mereka.

Untuk membatasi sebuah instance

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/>, lalu masuk ke akun penjual Anda.
2. Buka halaman [Produk Server](#), dan pada tab Produk server saat ini, pilih produk yang ingin Anda modifikasi.
3. Dari menu tarik-turun Permintaan perubahan, pilih Batasi instance.
4. Pilih contoh yang ingin Anda batasi, dan pilih Batasi.
5. Pilih Kirim permintaan perubahan untuk mengirimkan permintaan Anda untuk ditinjau.
6. Verifikasi bahwa tab Permintaan menampilkan status Permintaan sebagai Dalam peninjauan. Ketika permintaan selesai, status menjadi Berhasil.

Note

Jika kotak centang diarsir, ini berarti instance dikaitkan dengan satu hingga beberapa versi sebagai jenis instance yang direkomendasikan. Untuk membatasi instance tersebut, gunakan versi Perbarui untuk memilih jenis instans yang direkomendasikan berbeda. Setelah permintaan perubahan selesai dan instance yang ingin Anda batasi tidak lagi menjadi tipe instans yang direkomendasikan, Anda dapat kembali ke Batasi instance untuk membatasi instance yang Anda pilih.

Mengelola versi untuk produk berbasis AMI di AWS Marketplace

Saat Anda membuat produk berbasis Amazon Machine Image (AMI) AWS Marketplace, Anda menyertakan versi tertentu dari perangkat lunak Anda. Siklus hidup produk berbasis AMI untuk AWS Marketplace tidak berakhir setelah Anda mempublikasikan versi pertama. Anda harus menyimpan produk Anda up-to-date dengan versi baru perangkat lunak Anda. Bagian berikut menunjukkan cara mengelola versi Anda, yang mencakup memperbarui informasi versi (seperti deskripsi dan tanggal), menambahkan versi baru, dan membatasi akses ke versi sebelumnya.

Topik

- [Perbarui informasi versi](#)
- [Tambahkan versi baru](#)
- [Batasi versi](#)

Perbarui informasi versi

Setelah versi dibuat, dapat membantu untuk memberikan informasi terbaru kepada pembeli Anda dengan memodifikasi informasi yang terkait dengan versi. Misalnya, jika Anda berencana untuk membatasi versi 1.0 setelah versi 1.1 dirilis, Anda dapat memperbarui deskripsi versi 1.0 untuk mengarahkan pembeli ke versi 1.1, dengan tanggal versi tersebut akan dibatasi. Anda memperbarui informasi versi dari file Portal Manajemen AWS Marketplace.

Untuk memperbarui informasi versi

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/>, lalu masuk ke akun penjual Anda.
2. Buka halaman [produk server saat ini](#), pada tab Produk Server, lalu pilih produk yang ingin Anda modifikasi.
3. Dari menu menurun Minta perubahan, pilih Perbarui informasi versi.
4. Pada halaman Perbarui versi, pilih versi yang ingin Anda perbarui.
5. Perbarui salah satu informasi berikut yang perlu Anda modifikasi:
 - Catatan rilis
 - Instruksi penggunaan
 - Amazon Machine Image (AMI) 64 bit (x86) – Detail tentang penggunaan dan grup keamanan
6. Pilih Kirim.
7. Verifikasi bahwa permintaan muncul di tab Permintaan dengan status Dalam tinjauan.

Note

Anda tidak dapat menggunakan prosedur ini untuk memperbarui judul versi, atau AMI yang terkait dengan versi. Sebagai gantinya, [buat versi baru](#) dan [batasi versi sebelumnya](#).

Anda dapat memeriksa status permintaan Anda kapan saja dari tab [Permintaan](#) halaman [Produk Server](#). Untuk informasi selengkapnya, lihat [Dapatkan status permintaan perubahan](#).

Tambahkan versi baru

Anda dapat menambahkan versi baru dari produk Anda saat Anda membuat perubahan pada produk, citra dasar, atau saat lain kali Anda perlu memodifikasi AMI untuk produk. Tambahkan versi baru produk Anda dari Portal Manajemen AWS Marketplace.

Note

Untuk informasi tentang membuat AMI AWS Marketplace, lihat [Praktik terbaik untuk membangun AMIs untuk digunakan dengan AWS Marketplace](#).

Untuk menambahkan versi baru

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/>, lalu masuk ke akun penjual Anda.
2. Buka halaman [Produk Server](#), pada tab Produk server saat ini, lalu pilih produk yang ingin Anda modifikasi.
3. Dari menu menurun Perubahan permintaan, pilih Tambahkan versi baru. Formulir Tambahkan versi baru muncul, diisi dengan informasi dari versi terbaru Anda.

Note

Anda juga dapat memilih Uji 'Tambahkan versi' untuk memindai AMI dan AWS CloudFormation template Anda tanpa menambahkan versi baru. Untuk informasi selengkapnya, lihat [Memindai AMI Anda untuk persyaratan penerbitan](#).

4. Di bagian Informasi versi, berikan informasi berikut:
 - Judul versi - Masukkan string yang valid (misalnya *1.1* atau *Version 2.0*). Ini harus unik di seluruh produk.
 - Catatan rilis – Masukkan teks untuk menguraikan detail tentang versi ini.
5. Di bagian Opsi pengiriman, pilih bagaimana pembeli dapat menerapkan solusi, baik AMI (mandiri), AMI dengan CloudFormation, atau keduanya. Anda dapat memilih hingga tiga AMI dengan opsi CloudFormation pengiriman.

 Note

Opsi pengiriman baru tidak dapat ditambahkan ke versi yang sudah ada. Semua opsi pengiriman untuk satu versi harus dikirimkan dalam permintaan yang sama.

6. Di bagian Amazon Machine Image (AMI), berikan informasi berikut:
 - ID Amazon Machine Image – Masukkan ID AMI untuk AMI yang ingin Anda gunakan untuk versi ini. Anda dapat menemukan ID AMI [dari daftar AMIs di konsol](#). AMI harus ada di Wilayah AS Timur (Virginia N.), dan di akun AWS Marketplace Penjual Anda. Snapshot yang terkait dengan AMI ini tidak dapat dienkripsi.
 - Peran akses IAM ARN — Masukkan Nama Sumber Daya Amazon (ARN) untuk peran (IAM) AWS Identity and Access Management yang AWS Marketplace memungkinkan untuk mendapatkan akses ke AMI Anda. Untuk petunjuk tentang cara membuat IAM role, lihat [Memberikan AWS Marketplace akses ke AMI Anda](#). Gunakan format standar untuk ARN IAM, misalnya: `arn:aws:iam::123456789012:role/RoleName` ARN harus ada di akun AWS Marketplace Penjual Anda.
 - Sistem operasi (OS) — Masukkan sistem operasi dasar untuk AMI Anda.
 - Versi OS — Masukkan versi sistem operasi yang digunakan AMI Anda.
 - Nama pengguna OS — Untuk Linux berbasis AMIs, masukkan nama pengguna yang dapat digunakan untuk masuk ke instance. Kami merekomendasikan menggunakan `ec2-user`.
 - Port pemindaian — Masukkan nomor port yang dapat digunakan untuk masuk ke sistem operasi: port SSH untuk Linux AMI atau port RDP untuk AMI. Windows
7. Berikan konfigurasi berikut untuk bagian opsi pengiriman AMI, jika berlaku:
 - Tipe instans yang direkomendasikan – Pilih tipe instans yang pembeli dapatkan secara default.
 - Petunjuk penggunaan – Masukkan petunjuk untuk menggunakan AMI atau tautan ke informasi selengkapnya tentang menggunakan AMI. Misalnya: *To get started with the product, navigate to <https://example.com/usage.htm>.*
 - Titik akhir URL – Berikan informasi tentang bagaimana pembeli dapat mengakses perangkat lunak setelah mereka membuat sebuah instans. Masukkan Protokol (`https` atau `http`), URL Relatif (misalnya, `/index.html`), dan Port (misalnya, `443`) yang dapat digunakan pembeli untuk mengakses produk Anda. (Nama host tergantung pada EC2 instance, jadi Anda hanya perlu menyediakan jalur relatif).

- Rekomendasi grup keamanan — Masukkan informasi untuk satu atau lebih rekomendasi, termasuk protokol (TCP atau UDP), rentang port untuk mengizinkan, dan daftar IPv4 CIDR IPs (dalam bentuk xxx.xxx.xxx.xxx/nn, misalnya,). **192.0.2.0/24**
8. Berikan pengaturan konfigurasi berikut untuk AMI dengan opsi CloudFormation pengiriman, jika berlaku:
- Judul opsi pengiriman - Ini adalah nama unik untuk mengidentifikasi opsi pengiriman CloudFormation template AMI Tunggal ini.
 - Deskripsi Singkat - Sebuah deskripsi singkat dari CloudFormation template.
 - Deskripsi Panjang - Sebuah deskripsi rinci dari CloudFormation template.
 - Jenis Instance yang Direkomendasikan - Jenis instans, pembeli dapatkan secara default.
 - Instruksi penggunaan — Petunjuk untuk menggunakan solusi atau tautan ke informasi lebih lanjut tentang penggunaan solusi.
 - CloudFormation link template — URL ke lokasi CloudFormation template Anda di Amazon S3. Untuk tumpukan atau templat bersarang, berikan templat titik masuk, atau induk. Bucket Amazon S3 tempat file Anda berada harus dapat diakses publik. Untuk informasi selengkapnya, lihat [Menambahkan CloudFormation templat ke produk Anda](#).
 - Nama parameter AMI - Tambahkan nama parameter dalam CloudFormation template yang referensi sumber daya EC2 instans dalam template Anda untuk nilai ImageId propertinya. AWS Marketplace menggantikan parameter yang ditentukan dengan parameter [AWS Systems Manager Parameter Store](#). Saat pembeli menerapkan template Anda, parameter tersebut akan diselesaikan ke ID AWS AMI khusus Wilayah dari produk Anda yang dipublikasikan. Untuk informasi selengkapnya, lihat [Persyaratan untuk detail AMI](#).
 - Tautan diagram arsitektur — URL ke lokasi diagram arsitektur Anda di Amazon S3. File gambar maks harus berukuran 1560x878 atau 1560x3120 piksel dengan rasio 16:9 atau 1:2. Bucket Amazon S3 tempat file gambar Anda berada harus dapat diakses publik. Untuk persyaratan diagram, lihat [Diagram arsitektur](#).
9. Pilih Kirim untuk mengirimkan permintaan untuk menambahkan versi baru Anda.

 Note

Menambahkan versi baru menghasilkan pemindaian AMI. Untuk informasi selengkapnya, lihat [Memindai AMI Anda untuk persyaratan penerbitan](#).

10. Verifikasi bahwa permintaan muncul di tab Permintaan dengan status Dalam tinjauan. Jika ada kesalahan yang perlu diperbaiki, halaman akan menampilkan kesalahan dalam tabel di bagian atas halaman, dan bidang tertentu yang perlu diperbarui ditampilkan dalam warna merah.

Anda dapat memeriksa status permintaan Anda kapan saja dari tab Permintaan di halaman [Produk Server](#). Versi baru akan ditinjau dan, jika berhasil, dipublikasikan sebagai versi publik baru dari produk Anda. Jika ada masalah, statusnya mungkin diperlukan Tindakan. Pilih permintaan untuk melihat detail, termasuk masalah apa pun.

Jika permintaan Anda berhasil, pengguna yang ada akan menerima pesan email berikut. Pesan tersebut memberi tahu mereka bahwa versi baru tersedia, menautkan ke catatan rilis versi, dan menyarankan agar mereka meningkatkan ke versi terbaru. Sebagai pengguna Akun AWS root, Anda juga menerima salinan pesan email di akun email yang terkait dengan Anda Akun AWS.

```
Greetings from AWS Marketplace,
```

```
Thank you for subscribing to <product-title>
```

```
We are writing to inform you that <seller-name> has added a new version to <product-title> on AWS Marketplace.
```

```
As an existing customer, your subscription to the product, any running instances and access to previous versions are unaffected. However, <seller-name> does recommend you to update to the latest version, <product-title>/<version-title> by visiting <product-detail-page-of-new-listing>.
```

```
For additional questions or upgrade information, please contact <seller-name> directly. Click here <link of seller page on MP> to visit the seller's profile page on AWS Marketplace.
```

```
Release notes for <product-title>/<version-title>:
```

```
<release-notes>
```

```
Thank you,
```

```
The AWS Marketplace Team
```

```
https://aws.amazon.com/marketplace
```

```
AWS, Inc. is a subsidiary of Amazon.com, Inc. Amazon.com is a registered trademark of Amazon.com, Inc.
```

```
This message was produced and distributed by AWS Inc., 410 Terry Ave. North, Seattle, WA 98109-5210
```

Batasi versi

Jika Anda ingin mencegah pembeli mengakses versi tertentu dari produk publik Anda, Anda dapat membatasi versi tersebut.

Note

Semua pelanggan dapat menggunakan versi saat ini terlepas dari status pembatasan. AWS Marketplace pedoman mengharuskan Anda terus menawarkan dukungan kepada pembeli yang ada selama 90 hari setelah membatasi versi. AMI Anda akan ditandai sebagai usang setelah versi dibatasi. Untuk informasi selengkapnya, lihat [Menanggalkan AMI](#) di Panduan Pengguna Amazon Elastic Compute Cloud untuk Instans. Windows

Untuk membatasi versi

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/>, lalu masuk ke akun penjual Anda.
2. Buka halaman [Produk Server](#), pada tab Produk server saat ini, lalu pilih produk yang ingin Anda modifikasi.
3. Dari menu menurun Permintaan perubahan, pilih Batasi versi.
4. Pada halaman Batasi versi, pilih versi (atau beberapa versi) yang ingin Anda batasi.
5. Pilih Kirim untuk mengirimkan permintaan Anda untuk ditinjau.
6. Verifikasi bahwa tab Permintaan menampilkan status Permintaan sebagai Dalam peninjauan. Saat permintaan selesai, statusnya adalah Berhasil.

Note

Anda tidak dapat membatasi semua versi produk. Jika Anda mencoba untuk membatasi versi publik terakhir dari produk, Anda akan menerima kesalahan. Untuk menghapus produk sepenuhnya, lihat [the section called “Menghapus produk”](#).

Anda dapat memeriksa status permintaan Anda kapan saja dari tab Permintaan di halaman [Produk Server](#). Untuk informasi selengkapnya, lihat [Dapatkan status permintaan perubahan](#).

Note

Membatasi versi bisa memakan waktu hingga 3 hari untuk diselesaikan.

Jika permintaan Anda berhasil, pengguna Anda yang ada menerima pesan email berikut yang memberi tahu mereka tentang pembatasan versi dan menyarankan mereka menggunakan versi terbaru yang tersedia. Sebagai pengguna Akun AWS root, Anda juga menerima salinan pesan email di akun email yang terkait dengan Anda Akun AWS.

Greetings from AWS Marketplace,

Thank you for subscribing to <product-title>.

We are writing to inform you that, as of <Version-Restriction-Date>, <Seller Name> will no longer offer version(s) "<version-title>" to new subscribers. Your use and subscription is unaffected for this version(s), however it is recommended that users upgrade to the latest version on AWS Marketplace.

For additional questions or upgrade information, please contact <seller-name> directly. Click here<link of seller page on MP> to visit the seller's profile page on AWS Marketplace.

Thank you,

The AWS Marketplace Team

<https://aws.amazon.com/marketplace>

AWS, Inc. is a subsidiary of Amazon.com, Inc. Amazon.com is a registered trademark of Amazon.com, Inc. This message was produced and distributed by AWS Inc., 410 Terry Ave. North, Seattle, WA 98109-5210

Memperbarui informasi produk berbasis AMI tentang AWS Marketplace

Setelah membuat produk Amazon Machine Image (AMI) tunggal, Anda dapat mengubah beberapa informasi yang terkait dengannya AWS Marketplace. Misalnya, jika versi baru memodifikasi deskripsi atau sorotan produk, Anda dapat mengedit informasi produk dengan data baru. Anda juga dapat memperbarui informasi produk lainnya, termasuk judul produk, deskripsi SKU, kategori, kata kunci, dan lainnya. Untuk informasi lebih lanjut, lihat prosedur berikut.

Untuk memperbarui informasi produk

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/>, lalu masuk ke akun penjual Anda.
2. Buka halaman [Produk Server](#), dan pada tab Produk Server, pilih produk yang ingin Anda modifikasi.
3. Dari menu menurun Minta perubahan, pilih Perbarui informasi produk.
4. Perbarui salah satu bidang berikut yang perlu Anda ubah:
 - Judul produk
 - SKU
 - Deskripsi singkat
 - Deskripsi panjang
 - URL gambar logo produk
 - Sorotan
 - Kategori produk
 - Kata kunci
 - URL video produk
 - Sumber Daya
 - Informasi Support

 Note

Untuk detail tentang format logo, lihat [Persyaratan logo perusahaan dan produk](#).

5. Pilih Kirim.
6. Verifikasi bahwa permintaan muncul di tab Permintaan dengan status Dalam tinjauan. Anda mungkin perlu me-refresh halaman untuk melihat permintaan pada daftar.

Anda dapat memeriksa status permintaan Anda kapan saja dari tab Permintaan halaman [Produk Server](#). Untuk informasi selengkapnya, lihat [Dapatkan status permintaan perubahan](#).

Mengelola ketersediaan produk berbasis AMI menurut Wilayah AWS dan negara

Saat Anda membuat produk AWS Marketplace, Anda memilih AWS Wilayah di mana produk tersebut tersedia. Anda juga memilih negara tempat pembeli dapat membeli produk Anda. Kedua properti ini serupa, tetapi tidak sama. Misalnya, pembeli mungkin berlokasi di, dan membeli dari, Amerika Serikat, tetapi mereka mungkin berencana untuk menginstal produk Anda di Wilayah Eropa (Frankfurt). Agar pembeli ini dapat membeli produk Anda, Anda harus memasukkan Amerika Serikat dalam daftar negara Anda, dan Wilayah Eropa (Frankfurt) dalam daftar Wilayah Anda. Anda dapat menggunakan bagian berikut untuk memperbarui ketersediaan produk menurut Wilayah dan negara.

Topik

- [Menambahkan Wilayah AWS](#)
- [Membatasi Wilayah AWS](#)
- [Perbarui dukungan untuk masa depan Wilayah AWS](#)
- [Perbarui ketersediaan menurut negara](#)

Menambahkan Wilayah AWS

Anda dapat menambahkan Wilayah di mana pembeli dapat menggunakan produk Anda.

Untuk menambahkan Wilayah

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/>, lalu masuk ke akun penjual Anda.
2. Buka halaman [Produk Server](#), dan pada tab Produk server saat ini, pilih produk yang ingin Anda modifikasi.
3. Dari menu tarik-turun Permintaan perubahan, pilih Tambah Wilayah.
4. Pilih Wilayah yang ingin Anda tambahkan dari daftar Wilayah yang tersedia.
5. Pilih Kirim permintaan untuk mengirimkan permintaan Anda untuk ditinjau.
6. Verifikasi bahwa tab Permintaan menampilkan status Permintaan sebagai Dalam peninjauan. Ketika permintaan selesai, status menjadi Berhasil.

Note

Saat Anda menambahkan dukungan untuk yang baru Wilayah AWS, pelanggan yang sudah berlangganan penawaran pribadi untuk produk Anda tidak akan dapat mengakses Wilayah yang baru ditambahkan secara otomatis. Anda harus membuat penawaran pribadi lain dengan Wilayah yang ingin diakses pelanggan. Setelah menerima penawaran baru, pelanggan dapat mengakses Wilayah yang baru ditambahkan. Pelanggan yang berlangganan produk Anda di masa mendatang juga dapat mengakses Wilayah, selama Wilayah tersebut termasuk dalam penawaran pribadi. Untuk informasi selengkapnya tentang cara membuat penawaran pribadi baru, lihat [Peningkatan, pembaruan, dan amandemen penawaran pribadi](#).

Membatasi Wilayah AWS

Untuk mencegah pembeli baru menggunakan produk Anda secara spesifik Wilayah AWS, Anda dapat membatasi Wilayah. Anda dapat menambahkan Region kembali di lain waktu. Pelanggan produk yang ada di Wilayah dapat terus menggunakan produk dari Wilayah selama mereka berlangganan.

Untuk membatasi suatu Wilayah

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/>, lalu masuk ke akun penjual Anda.
2. Buka halaman [Produk Server](#), dan pada tab Produk server saat ini, pilih produk yang ingin Anda modifikasi.
3. Dari menu tarik-turun Permintaan perubahan, pilih Batasi Wilayah.
4. Pilih menu tarik-turun untuk melihat daftar Wilayah di mana produk Anda saat ini tersedia.
5. Pilih Wilayah yang ingin Anda batasi.
6. Wilayah yang Anda pilih muncul sebagai token. Tinjau daftar Wilayah yang Anda batasi, dan masukkan X untuk Wilayah yang tidak ingin Anda batasi.
7. Pilih Kirim permintaan perubahan untuk mengirimkan permintaan Anda untuk ditinjau.
8. Verifikasi bahwa tab Permintaan menampilkan status Permintaan sebagai Dalam peninjauan. Ketika permintaan selesai, status menjadi Berhasil.

Jika permintaan Anda berhasil, pengguna Anda yang ada akan menerima pesan email berikut yang memberi tahu mereka tentang Wilayah yang akan dibatasi. Mereka dapat terus menggunakan produk Anda selama mereka tetap berlangganan, tetapi mereka tidak dapat berlangganan ulang jika mereka membatalkan langganan.

Greetings from AWS Marketplace,

This message is a notification detailing a recent change for <ProductName>. {{{sellerName}}} has opted to restrict the <ProductType> product in <Restricted Region(s)> beginning <DateOfChange>.

This impacts you in the following ways:

1. As long as you're subscribed to the product, you can continue using the software product in the restricted Region.
2. You can't begin new instances of the software product in the restricted Region.
3. You can continue using the software product in all available AWS Regions.

Regards,

The AWS Marketplace Team

AWS, Inc. is a subsidiary of Amazon.com, Inc. Amazon.com (<http://amazon.com/>) is a registered

trademark of Amazon.com, Inc. This message was produced and distributed by Amazon Web Services Inc., 410 Terry Ave. North, Seattle, WA 98109-5210.

Perbarui dukungan untuk masa depan Wilayah AWS

Jika Anda ingin produk Anda di-onboard untuk yang baru diluncurkan Wilayah AWS, Anda dapat menggunakan dukungan Update future Region.

Untuk memperbarui dukungan Wilayah future

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/>, lalu masuk ke akun penjual Anda.
2. Buka halaman [Produk Server](#), dan pada tab Produk server saat ini, pilih produk yang ingin Anda modifikasi.
3. Dari menu tarik-turun Permintaan perubahan, pilih Perbarui dukungan Wilayah future.
4. Anda dapat memilih untuk mengaktifkan dukungan Wilayah future AWS Marketplace untuk memungkinkan onboard produk Anda yang baru diluncurkan Wilayah AWS atas nama Anda.

5. Setelah mengaktifkan fitur, Anda dapat memilih di antara semua Wilayah masa depan atau membatasi hanya untuk Wilayah AS.
6. Pilih Kirim permintaan perubahan untuk mengirimkan permintaan Anda untuk ditinjau.
7. Verifikasi bahwa tab Permintaan menampilkan status Permintaan sebagai Dalam peninjauan. Ketika permintaan selesai, status menjadi Berhasil.

Perbarui ketersediaan menurut negara

Jika Anda ingin mengubah negara tempat produk Anda dapat berlangganan dan ditawarkan, Anda dapat menggunakan Ketersediaan pembaruan.

Untuk memperbarui ketersediaan menurut negara

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/>, lalu masuk ke akun penjual Anda.
2. Buka halaman [Produk Server](#), pada tab Produk server saat ini, lalu pilih produk yang ingin Anda modifikasi.
3. Dari menu tarik-turun Permintaan perubahan, pilih Perbarui ketersediaan.
4. Pilih salah satu opsi berikut:
 1. Semua negara - Tersedia di semua negara yang didukung.
 2. Semua negara dengan pengecualian — Tersedia di semua negara yang didukung kecuali di negara tertentu.
 3. Daftar kustom - Daftar spesifik negara tempat produk tersedia.
5. Pilih Kirim permintaan perubahan untuk mengirimkan permintaan Anda untuk ditinjau.
6. Verifikasi bahwa tab Permintaan menampilkan status Permintaan sebagai Dalam peninjauan. Ketika permintaan selesai, status menjadi Berhasil.

Memperbarui perjanjian lisensi pengguna akhir (EULA) untuk produk berbasis AWS Marketplace AMI Anda

Sebagai AWS Marketplace penjual, Anda dapat memperbarui perjanjian lisensi pengguna akhir (EULA) yang akan mengatur penggunaan produk Amazon Machine Image (AMI) tunggal Anda. EULA Anda terletak di halaman daftar produk untuk daftar perangkat lunak publik di AWS Marketplace. Anda dapat menerapkan EULA Anda sendiri atau menggunakan [Kontrak Standar untuk AWS](#)

[Marketplace \(SCMP\)](#). Prosedur berikut menunjukkan cara memperbarui EULA untuk produk Single-AMI Anda.

Untuk informasi lebih lanjut tentang EULA, lihat [Menggunakan kontrak standar di AWS Marketplace](#).

Untuk memperbarui EULA

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/>, lalu masuk ke akun penjual Anda.
2. Pilih tab [Produk Server](#), pada tab Produk server saat ini, pilih produk yang ingin Anda modifikasi.
3. Dari menu tarik-turun Permintaan perubahan, pilih Perbarui perjanjian lisensi pengguna akhir.
4. Anda dapat memilih [Kontrak Standar untuk AWS Marketplace \(SCMP\)](#) atau mengirimkan EULA kustom Anda sendiri. Untuk EULA kustom, Anda harus memberikan URL untuk kontrak kustom Anda dari bucket Amazon S3.

 Note

Aksesibilitas publik harus diaktifkan di bucket Amazon S3 Anda.

5. Pilih Kirim permintaan perubahan untuk mengirimkan permintaan Anda untuk ditinjau.
6. Verifikasi bahwa tab Permintaan menampilkan status Permintaan sebagai Dalam peninjauan. Ketika permintaan selesai, status menjadi Berhasil.

Memperbarui kebijakan pengembalian dana untuk produk berbasis AMI Anda di AWS Marketplace

Sebagai AWS Marketplace penjual, Anda dapat mengatur kebijakan pengembalian dana untuk produk Amazon Machine Image (AMI) tunggal Anda. Jika Anda ingin mengubah kebijakan pengembalian dana untuk produk Anda, Anda dapat menggunakan kebijakan pengembalian dana Perbarui. Prosedur berikut menunjukkan cara memperbarui kebijakan pengembalian dana Anda.

Untuk memperbarui kebijakan pengembalian dana

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/>, lalu masuk ke akun penjual Anda.
2. Buka halaman [Produk Server](#), pada tab Produk server saat ini, lalu pilih produk yang ingin Anda modifikasi.

3. Dari menu tarik-turun Permintaan perubahan, pilih Perbarui kebijakan pengembalian dana.
4. Rincian kebijakan pengembalian dana saat ini disediakan di kotak teks untuk Anda edit. Mengirimkan permintaan menimpa kebijakan pengembalian dana saat ini.
5. Pilih Kirim permintaan perubahan untuk mengirimkan permintaan Anda untuk ditinjau.
6. Verifikasi bahwa tab Permintaan menampilkan status Permintaan sebagai Dalam peninjauan. Ketika permintaan selesai, status menjadi Berhasil.

Memberikan AWS Marketplace akses ke AMI Anda

Saat Anda membuat permintaan yang menyertakan penambahan Amazon Machine Image (AMI) baru AWS Marketplace, AMI harus disalin ke dalam AWS Marketplace sistem dan kemudian dipindai untuk masalah keamanan. Anda harus memberikan AWS Marketplace akses ke AMI dengan membuat peran AWS Identity and Access Management (IAM) dengan izin untuk melakukan tindakan pada AMI Anda dan kebijakan kepercayaan yang memungkinkan AWS Marketplace untuk mengambil peran tersebut. Anda hanya perlu membuat IAM role satu kali. Prosedur berikut menunjukkan kepada Anda cara membuat peran untuk konsumsi AWS Marketplace aset yang memberikan AWS Marketplace akses ke AMI Anda.

Untuk membuat peran untuk konsumsi aset AWS Marketplace AMI

1. Masuk ke AWS Management Console, buka konsol IAM dan buka [halaman Peran](#).
2. Pilih Buat peran.
3. Pada halaman Buat peran, buat pilihan berikut:
 - Pilih jenis entitas tepercaya — Pilih AWS Layanan.
 - Pilih kasus penggunaan — Pilih AWS Marketplace.
 - Pilih kasus penggunaan Anda – Pilih Marketplace – Penyerapan Aset AMI.
 - Untuk pindah ke halaman berikutnya, pilih Berikutnya: Izin.
4. Pilih AWSPMarketplaceAmiIngestionkebijakan. Tambahkan batas izin jika diperlukan, kemudian pilih Berikutnya: Tag untuk melanjutkan.

Note

Anda dapat menggunakan batas izin untuk membatasi akses yang Anda berikan kepada AWS Marketplace dengan peran ini. Untuk informasi selengkapnya, lihat [Batas izin untuk entitas IAM](#) dalam Panduan Pengguna.

5. Untuk melanjutkan, pilih Berikutnya: Tinjau.
6. Berikan nama untuk peran, kemudian pilih Buat peran.
7. Anda akan melihat “Peran *rolename* telah dibuat” di bagian atas halaman, dan peran akan muncul dalam daftar peran.

Pada halaman ini, saat Anda memilih peran yang baru saja Anda buat, Anda dapat melihat ARN - nya dalam formulir `arn:aws:iam::123456789012:role/exampleRole`. Gunakan ARN untuk ARN peran akses IAM saat Anda membuat permintaan perubahan, misalnya, saat [menambahkan versi baru](#) ke produk Anda.

Menghapus produk dari AWS Marketplace

Setelah Anda mempublikasikan produk Anda, Anda dapat menghapusnya AWS Marketplace. Ini juga disebut sebagai matahari terbenam. Untuk menghapus produk, identifikasi produk dan kirimkan permintaan untuk menghapusnya, bersama dengan alasan penghapusan dan alamat email kontak untuk Anda. Anda juga dapat memberikan ID produk pengganti jika Anda mengganti produk saat ini dengan yang baru.

Setelah Anda meminta penghapusan produk, pelanggan baru tidak dapat berlangganan produk. Anda harus mendukung pelanggan yang sudah ada selama minimal 90 hari.

Note

Anda tidak dapat menghapus produk yang dibatasi dari Portal AWS Marketplace Manajemen. Portal menyimpannya sebagai bagian dari riwayat publikasi akun.

Kami memproses permintaan penghapusan produk AWS Marketplace dengan ketentuan sebagai berikut:

- Produk dihapus dari AWS Marketplace pencarian, penelusuran, dan alat penemuan lainnya. Tombol atau fungsionalitas Berlangganan apa pun dinonaktifkan, dan pesan di halaman dengan jelas menunjukkan bahwa produk tidak lagi tersedia. Halaman detail produk masih dapat diakses menggunakan URL dan dapat diindeks di mesin pencari publik.
- Alasan penghapusan harus ditentukan (misalnya, akhir dukungan, akhir pembaruan produk, atau produk pengganti). Untuk persyaratan untuk melanjutkan dukungan untuk produk yang dihapus, lihat [Syarat dan Ketentuan untuk AWS Marketplace Penjual](#).
- AWS Marketplace menghubungi pembeli saat ini melalui pesan email yang memberi tahu mereka tentang penghapusan produk, alasan penghapusan, dan untuk memberikan informasi kontak penjual.
- Pembeli saat ini mempertahankan akses ke perangkat lunak hingga mereka membatalkan langganan mereka. Mereka tidak terpengaruh dengan cara apa pun oleh penghapusan produk.

Untuk menghapus produk yang dibuat menggunakan Portal Manajemen AWS Marketplace

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/>, lalu masuk ke akun penjual Anda.
2. Pilih **Produk** tab, kemudian pilih **Server**.
3. Pada halaman produk Anda, di bawah **Produk Server**, temukan produk yang ingin Anda hapus. Dari daftar tarik-turun **Permintaan perubahan**, pilih **Perbarui visibilitas produk**.
4. Pada halaman **Perbarui visibilitas produk**, pilih **Dibatasi**.
5. (Opsional) Sediakan ID Produk Penggantian, jika ada produk lain yang akan menggantikan produk yang Anda hapus.
6. Tinjau informasi untuk akurasi, lalu pilih **Kirim**.

Apakah selanjutnya menampilkan halaman informasi setelah Anda mengirimkan permintaan penghapusan produk. Operasi AWS Marketplace Penjual meninjau dan memproses permintaan Anda. Periksa status kiriman Anda dengan melihat **Permintaan**.

Setelah produk Anda dihapus, produk akan muncul di daftar **Produk Saat Ini** di Portal Manajemen AWS Marketplace. Masuk **Produk** saat ini, satu-satunya tindakan yang dapat Anda lakukan adalah mengunduh spreadsheet untuk produk tersebut. Anda tidak dapat mengedit atau mengirimkan permintaan matahari terbenam lainnya.

Jika Anda memiliki pertanyaan tentang penghapusan produk, hubungi [AWS Marketplace Tim Operasional Penjual](#).

Memecahkan masalah kesalahan umum untuk permintaan perubahan pada AWS Marketplace

Ketika Anda membuat perubahan pada informasi produk Anda AWS Marketplace, Anda mungkin mengalami kesalahan. Topik ini menjelaskan beberapa kesalahan umum dan memberikan saran tentang cara memperbaikinya.

- Memindai Anda AMI — Beberapa masalah dapat terjadi saat memindai AMI:
 - Anda belum memberikan AWS Marketplace izin untuk memindai AMI. Hibah AWS Marketplace izin untuk mengaksesnya. Atau Anda telah memberikan izin, tetapi batas izin terlalu ketat. Untuk informasi selengkapnya, lihat [Memberikan AWS Marketplace akses ke AMI Anda](#).
 - Jika pemindaian menemukan masalah keamanan atau Kerentanan Umum dan Eksposur (CVEs) di Anda AMI, pastikan Anda menggunakan tambalan terbaru untuk sistem operasi di gambar Anda. Untuk informasi selengkapnya, lihat [Persyaratan produk berbasis AMI untuk AWS Marketplace](#).

Untuk pedoman umum tentang membangun AMI, lihat [Praktik terbaik untuk membangun AMIs untuk digunakan dengan AWS Marketplace](#).

- Portal Manajemen AWS Marketplace bidang — Beberapa bidang di Portal Manajemen AWS Marketplace memerlukan informasi yang sangat spesifik:
 - Jika Anda tidak yakin tentang apa bidang meminta, coba periksa detail di konsol. Sebagian besar bidang memiliki deskripsi teks di atas lapangan, dan persyaratan format di bawah lapangan.
 - Jika Anda mencoba mengirimkan formulir dengan satu atau lebih bidang yang tidak valid, daftar masalah akan ditampilkan. Tindakan yang disarankan diberikan untuk membantu Anda memperbaiki masalah.
 - Jika Anda diminta untuk memberikan ARN, Anda biasanya akan menemukannya di tempat lain di konsol. Misalnya, ARN untuk IAM peran yang Anda buat untuk memberikan AWS Marketplace akses ke Anda AMI ditemukan di [halaman Peran](#) di IAM konsol. ARNs Semua memiliki format yang sama. Misalnya, IAM peran ARN dalam bentuk `arn:aws:iam: :123456789012:role/exampleRole`
 - Logo dan video Anda harus disediakan secara URL langsung ke konten. Untuk informasi selengkapnya tentang format logo, lihat [Persyaratan logo perusahaan dan produk](#).

Untuk informasi selengkapnya tentang mengirimkan produk dan permintaan perubahan versi, lihat [Mengirimkan produk Anda untuk dipublikasikan AWS Marketplace](#).

- Masalah Formulir Beban Produk (PLF) - PLFs berisi instruksi yang disertakan dalam spreadsheet. Petunjuk keseluruhan disediakan dalam tabel Petunjuk. Setiap bidang memiliki petunjuk tentang cara mengisinya, pilih bidang untuk mengungkapkan petunjuk.
- Permintaan dalam Proses — Beberapa permintaan tidak dapat terjadi secara paralel. Anda hanya dapat memiliki satu permintaan untuk memperbarui informasi spesifik yang sedang berlangsung untuk suatu produk pada satu waktu. Anda dapat melihat semua permintaan Anda masih dalam peninjauan pada tab Permintaan pada halaman Produk Server di Portal Manajemen AWS Marketplace. Jika Anda memiliki permintaan tertunda yang tidak Anda inginkan, Anda dapat membatalkannya kemudian mengirimkan permintaan baru dengan perubahan yang ingin Anda buat.
- Anda tidak dapat memperbarui informasi versi saat pembaruan (untuk menambah atau membatasi) versi sedang berlangsung.
- Jika ada permintaan yang tertunda dari tim Operasi AWS Marketplace Penjual, Anda tidak dapat mengirimkan perubahan baru.
- Kesalahan tidak dapat dijelaskan— Jika kiriman Anda gagal tanpa penjelasan, coba lagi. Kadang-kadang, beban server menyebabkan pengiriman gagal.

Jika Anda masih mengalami masalah dengan permintaan perubahan, hubungi tim [Operasi AWS Marketplace Penjual](#).

Praktik terbaik untuk membangun AMIs untuk digunakan dengan AWS Marketplace

Topik ini memberikan praktik dan referensi terbaik untuk membantu Anda membuat Amazon Machine Images (AMIs) untuk digunakan AWS Marketplace. AMIs dibangun dan diserahkan AWS Marketplace harus mematuhi semua kebijakan AWS Marketplace produk. Untuk informasi selengkapnya, silakan lihat bagian-bagian berikut ini.

Topik

- [Mengamankan hak jual kembali](#)
- [Membangun AMI](#)
- [Mempersiapkan dan mengamankan AMI Anda AWS Marketplace](#)

- [Memindai AMI Anda untuk persyaratan penerbitan](#)
- [Memverifikasi perangkat lunak Anda berjalan di perangkat AWS Marketplace AMI](#)

Mengamankan hak jual kembali

Untuk distribusi Linux yang tidak bebas, Anda bertanggung jawab untuk mengamankan hak jual kembali untuk mereka dengan pengecualian AWS Amazon Linux, RHEL, dan SUSE yang disediakan. Anda tidak perlu mengamankan hak jual kembali untuk Windows AMIs.

Membangun AMI

Gunakan panduan berikut untuk membangun AMIs:

- Pastikan AMI Anda memenuhi semua [AWS Marketplace kebijakan](#).
- Buat AMI Anda di US East (N. Virginia).
- Buat produk dari yang sudah ada dan terpelihara dengan baik yang AMIs didukung oleh Amazon Elastic Block Store (Amazon EBS) dengan siklus hidup yang ditentukan dengan jelas yang disediakan oleh sumber tepercaya dan bereputasi baik seperti. AWS Marketplace
- Bangun AMIs menggunakan sebagian besar sistem up-to-date operasi, paket, dan perangkat lunak.
- Pastikan AMI Anda didasarkan pada Amazon EC2 AMI publik, yang menggunakan virtualisasi hardware virtual machine (HVM) dan arsitektur 64-bit.
- Kembangkan proses berulang untuk membangun, memperbarui, dan menerbitkan ulang AMIs.
- Gunakan nama pengguna sistem operasi (OS) yang konsisten di semua versi dan produk. Nama pengguna default yang direkomendasikan adalah `ec2-user` untuk Linux dan sistem mirip Unix lainnya, dan `Administrator` untuk Windows.
- Sebelum mengirimkan AMI final ke AWS Marketplace penerbitan, luncurkan dan uji instans dari AMI Anda untuk memverifikasi pengalaman pengguna akhir yang dimaksud. Uji semua metode instalasi, fitur, dan kinerja pada instance ini.
- Periksa pengaturan port sebagai berikut:
 - Sebagai [konfigurasi keamanan praktik terbaik](#) terhadap firewall terbuka, proxy terbalik, dan kerentanan SSRF, opsi dukungan IMDS harus disetel hanya. IMDSv2 CLI berikut dapat digunakan saat mendaftarkan AMI baru pada tahap pembuatan akhir:
 - `aws ec2 register-image --name my-image --root-device-name /dev/xvda --block-device-mappings DeviceName=/dev/`

```
xvda,Ebs={SnapshotId=snap-0123456789example} --architecture x86_64 --  
imds-support v2.0
```

Untuk informasi selengkapnya tentang izin dan tentang pembuatan kebijakan, lihat sumber daya berikut:

- [Buat AMI yang didukung Amazon EBS-backed di Panduan Pengguna Amazon EC2](#)
- [Buat Amazon EC2 AMI menggunakan Windows Sysprep di Panduan Pengguna Amazon EC2](#)
- [Bagaimana cara membuat Amazon Machine Image \(AMI\) dari instans yang didukung EBS?](#)
- [Amazon Linux AMI](#)
- [Jenis EC2 Instans Amazon dan Jenis Instance](#)
- [Mengkonfigurasi AMI untuk penggunaan IMDS V2 secara default](#)

Mempersiapkan dan mengamankan AMI Anda AWS Marketplace

Kami merekomendasikan panduan berikut untuk membuat aman AMIs:

- Gunakan [Pedoman untuk Linux Bersama AMIs](#) di Panduan EC2 Pengguna Amazon
- Arsitek AMI Anda untuk men-deploy sebagai instalasi minimum untuk mengurangi permukaan serangan. Nonaktifkan atau hapus layanan dan program yang tidak perlu.
- Bila memungkinkan, gunakan end-to-end enkripsi untuk lalu lintas jaringan. Misalnya, gunakan Secure Sockets Layer (SSL) untuk mengamankan sesi HTTP antara Anda dan pembeli Anda. Pastikan bahwa layanan Anda hanya menggunakan up-to-date sertifikat dan valid.
- Saat mendokumentasikan produk AMI Anda, berikan rekomendasi grup keamanan bagi pembeli untuk mengontrol akses lalu lintas masuk ke instans mereka. Rekomendasi Anda harus menentukan yang berikut:
 - Set minimum port yang diperlukan agar layanan Anda berfungsi.
 - Port yang direkomendasikan dan rentang alamat IP sumber untuk akses administratif.

Rekomendasi grup keamanan ini membantu pembeli menerapkan kontrol akses yang tepat. Untuk informasi selengkapnya tentang cara menambahkan versi baru untuk produk AMI Anda, lihat [Tambahkan versi baru](#).

- Pertimbangkan untuk melakukan tes penetrasi terhadap lingkungan AWS komputasi Anda secara berkala, atau pertimbangkan untuk mempekerjakan pihak ketiga untuk melakukan pengujian

tersebut atas nama Anda. Untuk informasi selengkapnya, termasuk formulir permintaan pengujian penetrasi, lihat [AWS Pengujian penetrasi](#).

- Sadarilah 10 kerentanan teratas untuk aplikasi web, dan bangun aplikasi Anda sesuai dengan itu. Untuk mempelajari lebih lanjut, lihat [Proyek Keamanan Aplikasi Web Terbuka \(OWASP\) - Top 10 Web Aplikasi Risiko Keamanan](#). Saat kerentanan internet baru ditemukan, segera memperbarui aplikasi web yang dikirimkan di AMI Anda. Contoh sumber daya yang mencakup informasi ini adalah [SecurityFocus](#) dan Database [Kerentanan Nasional NIST](#).

Untuk informasi selengkapnya terkait keamanan, lihat sumber daya berikut:

- [AWS Cloud Keamanan](#)
- [Pusat Keamanan Internet \(CIS\): Tolok Ukur Keamanan](#)
- [Proyek Keamanan Aplikasi Web Terbuka \(OWASP\): Praktik Pengkodean Aman - Panduan Referensi Cepat](#)
- [OWASP 10 Risiko Keamanan Aplikasi Web Teratas](#)
- [SANS \(SysAdmin, Audit, Jaringan, dan Keamanan\) Pencacahan Kelemahan Umum \(CWE\) 25 Kesalahan Perangkat Lunak Paling Berbahaya](#)
- [Fokus Keamanan](#)
- [Database Kerentanan Nasional NIST](#)

Memindai AMI Anda untuk persyaratan penerbitan

Untuk mempublikasikan AMI Anda di AWS Marketplace Katalog, Anda harus menyelesaikan pemindaian AMI. Pemindaian AMI memeriksa kerentanan dan eksposur umum yang belum ditambal (CVEs) dan memverifikasi bahwa AMI Anda mengikuti praktik terbaik keamanan. Untuk informasi selengkapnya, lihat [Mempersiapkan dan mengamankan AMI Anda AWS Marketplace](#)

Untuk melakukan pemindaian AMI, pilih salah satu opsi berikut:

Opsi 1: Menu Aset

Metode ini memungkinkan pemindaian AMIs di luar aliran pembuatan produk. Ini juga berguna untuk penjual SaaS yang menggunakan SAAS Quick Launch yang perlu memindai aset tanpa membuat produk AMI.

1. Dari menu [Portal Manajemen AWS Marketplace](#), navigasikan ke menu Aset dan pilih Amazon Machine Image.

2. Untuk memulai proses pemindaian, pilih Tambahkan AMI.
3. Anda dapat melihat status AMIs pemindaian dengan kembali ke halaman ini.

Opsi 2: Minta perubahan menu

Opsi ini tersedia untuk penjual yang telah membuat produk AMI. Pelajari lebih lanjut di [Membuat produk berbasis AMI](#)

1. Dari [Portal Manajemen AWS Marketplace](#), navigasikan ke menu Produk dan pilih Server.
2. Pilih produk Anda dari produk Server. Ini harus menjadi produk berbasis AMI. Produk dapat berada di negara bagian mana pun dan tidak perlu berada dalam keadaan Publik yang diterbitkan untuk langkah selanjutnya.
3. Arahkan ke menu Permintaan perubahan dan pilih Perbarui versi.
4. Pilih Uji 'Tambahkan versi'. Ikuti petunjuk untuk mengirimkan permintaan dengan detail AMI Anda. Jika permintaan berhasil, ini menunjukkan bahwa AMI telah berhasil melewati pemindaian. Berbeda dengan opsi Tambahkan versi baru, Uji 'tambahkan versi' tidak menambahkan versi baru ke produk berbasis AMI jika pemindaian berhasil.

 Note

Untuk mempelajari tentang memberikan AWS Marketplace akses ke AMI Anda, lihat [Memberikan AWS Marketplace akses ke AMI Anda](#).

Memverifikasi perangkat lunak Anda berjalan di perangkat AWS Marketplace AMI

Anda mungkin ingin agar perangkat lunak Anda memverifikasi saat runtime bahwa perangkat lunak tersebut berjalan di EC2 instans Amazon yang dibuat dari produk AMI Anda.

Untuk memverifikasi EC2 instans Amazon dibuat dari produk AMI Anda, gunakan layanan metadata instans yang ada di Amazon. EC2 Langkah-langkah berikut membawa Anda melalui validasi ini.

Untuk informasi selengkapnya tentang penggunaan layanan metadata, lihat [Metadana instance dan data pengguna](#) di Panduan Pengguna Amazon Elastic Compute Cloud.

1. Dapatkan dokumen identitas instance

Setiap contoh berjalan memiliki dokumen identitas yang dapat diakses dari contoh yang menyediakan data tentang contoh itu sendiri. Contoh berikut menunjukkan menggunakan curl dari contoh untuk mengambil dokumen identitas contoh.

IMDSv2: (Direkomendasikan)

```
TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600" ` \
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/dynamic/instance-identity/document
{
  "accountId" : "0123456789",
  "architecture" : "x86_64",
  "availabilityZone" : "us-east-1e",
  "billingProducts" : null,
  "devpayProductCodes" : null,
  "marketplaceProductCodes" : [ "0vg0000000000000000000000000000" ],
  "imageId" : "ami-0123456789abcdef1",
  "instanceId" : "i-0123456789abcdef0",
  "instanceType" : "t2.medium",
  "kernelId" : null,
  "pendingTime" : "2020-02-25T20:23:14Z",
  "privateIp" : "10.0.0.2",
  "ramdiskId" : null,
  "region" : "us-east-1",
  "version" : "2017-09-30"
}
```

IMDSv1:

```
curl http://169.254.169.254/latest/dynamic/instance-identity/document{
  "accountId" : "0123456789",
  "architecture" : "x86_64",
  "availabilityZone" : "us-east-1e",
  "billingProducts" : null,
  "devpayProductCodes" : null,
  "marketplaceProductCodes" : [ "0vg0000000000000000000000000000" ],
  "imageId" : "ami-0123456789abcdef1",
  "instanceId" : "i-0123456789abcdef0",
  "instanceType" : "t2.medium",
  "kernelId" : null,
```

```
"pendingTime" : "2020-02-25T20:23:14Z",  
"privateIp" : "10.0.0.2",  
"ramdiskId" : null,  
"region" : "us-east-1",  
"version" : "2017-09-30"  
}
```

2. Verifikasi dokumen identitas instance

Anda dapat memverifikasi bahwa identitas contoh benar menggunakan tanda tangan. Untuk detail tentang proses ini, lihat [Dokumen identitas instance](#) di Panduan Pengguna Amazon Elastic Compute Cloud.

3. Verifikasi kode produk

Saat Anda awalnya mengirimkan produk AMI Anda untuk dipublikasikan, produk Anda diberi [Kode Produk](#) oleh AWS Marketplace. Anda dapat memverifikasi kode produk dengan memeriksa `marketplaceProductCodes` di dokumen identitas contoh, atau Anda bisa mendapatkannya langsung dari layanan metadata:

IMDSv2:

```
TOKEN=`curl -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-metadata-token-ttl-seconds: 21600" ` \  
&& curl -H "X-aws-ec2-metadata-token: $TOKEN" http://169.254.169.254/latest/meta-data/product-codes
```

Jika kode produk cocok dengan satu untuk produk AMI Anda, maka instans dibuat dari produk Anda.

Anda juga dapat memverifikasi informasi lain dari dokumen identitas instans, seperti `instanceId` dan `instancePrivateIp`.

Harga produk AMI untuk AWS Marketplace

AWS Marketplace memiliki beberapa model harga untuk produk Amazon Machine Image (AMI). Dengan penawaran pribadi penjual, ada opsi yang tersedia untuk kontrak multi-tahun dan durasi khusus. Untuk informasi selengkapnya tentang kontrak multi-tahun dan durasi khusus, lihat [Mempersiapkan penawaran pribadi untuk AWS Marketplace produk Anda](#) dan [Paket cicilan](#)

[penawaran pribadi](#). Bagian berikut memberikan informasi tentang model harga untuk produk berbasis AMI.

Note

Anda harus dapat memberikan formulir pajak W-9 (untuk entitas yang berbasis di AS) atau formulir W-8 (untuk entitas berbasis UE) seperti yang dijelaskan dalam [Mendaftar sebagai AWS Marketplace penjual](#)

Topik

- [Model harga AMI](#)
- [AWS biaya dan biaya perangkat lunak](#)
- [Harga pengukuran khusus untuk produk AMI dengan AWS Marketplace Metering Service](#)
- [Harga kontrak untuk produk AMI pada AWS Marketplace](#)
- [Mengaitkan lisensi dengan produk berbasis AMI menggunakan AWS License Manager](#)

Model harga AMI

Tabel berikut memberikan informasi umum tentang model harga untuk produk berbasis AMI.

Model penentuan harga	Deskripsi
Kosong	<p>Pelanggan dapat menjalankan instans sebanyak yang didukung Amazon Elastic Compute Cloud EC2 (Amazon) tanpa biaya perangkat lunak tambahan yang dikeluarkan.</p> <div data-bbox="669 1476 711 1514" style="float: left; margin-right: 5px;"></div> <div data-bbox="716 1476 795 1514" style="float: left;">Note</div> <div data-bbox="716 1530 1408 1619" style="clear: both;"> <p>Uji Coba Gratis dan Harga Tahunan tidak dapat digabungkan dengan harga Bulanan.</p> </div>
Bawa lisensi Anda sendiri (BYOL)	AWS Marketplace tidak membebankan biaya kepada pelanggan untuk penggunaan perangkat lunak, tetapi pelanggan harus menyediakan kunci lisensi untuk mengaktifkan produk. Kunci ini dibeli di luar AWS Marketplace.

Model penentuan harga	Deskripsi
	Penegakan hak dan lisensi, selain semua harga dan penagihan, ditangani oleh Anda.

Model penentuan harga	Deskripsi
Dibayar per jam atau per jam-tahunan	<p>Per Jam — Perangkat lunak dibebankan per jam. Setiap jenis instance dapat diberi harga berbeda (tetapi tidak harus), dan penggunaan dibulatkan ke satu jam penuh terdekat.</p> <p>Setiap Jam dengan Uji Coba Gratis — Pelanggan dibatasi untuk menjalankan persis satu contoh perangkat lunak tanpa dikenakan biaya. Anda menentukan durasi, antara 5 dan 30 hari. Uji coba gratis berlaku untuk jenis instans paling mahal yang sedang berjalan, dan setiap penggunaan bersamaan di luar instans 1 ditagih dengan tarif per jam. CATATAN: Ini adalah model yang berbeda dari Tingkat AWS Gratis untuk EC2 penggunaan Amazon di mana pelanggan diberikan 750 jam penggunaan gratis setiap bulan.</p> <p>Per Jam dan Bulanan - Biaya per jam dan bulanan diterapkan secara independen. Biaya bulanan dibebankan setiap bulan terlepas dari penggunaan, dan biaya per jam diterapkan hanya berdasarkan penggunaan per jam.</p> <p>Setiap Jam dengan Tahunan — Pelanggan memiliki opsi untuk membeli penggunaan satu tahun di muka untuk satu EC2 instance Amazon dari satu jenis instans. Anda menetapkan harga untuk setiap jenis instans dan dapat menawarkan penghematan bersih selama harga per jam. Setiap penggunaan pelanggan di atas jumlah langganan tahunan yang dibeli ditagih dengan tarif per jam yang Anda tetapkan untuk jenis instans tersebut.</p> <p>Setiap Jam dengan Durasi Multi-Tahunan dan Kustom - Jenis penawaran ini hanya tersedia melalui penawaran pribadi penjual. Menggunakan penawaran pribadi penjual, Anda menentukan durasi kontrak khusus, hingga 3 tahun. Anda dapat menentukan pembayaran di muka, atau menyertakan jadwal pembayaran yang fleksibel. Anda menetapkan harga untuk setiap jenis instans. Jika ada</p>

Model penentuan harga	Deskripsi
	<p>jadwal pembayaran yang fleksibel dalam penawaran, Anda juga menetapkan tanggal faktur, jumlah pembayaran, dan jumlah instans untuk setiap jenis instans yang termasuk dalam penawaran. Untuk penawaran pribadi penjual aktif dengan jadwal pembayaran yang fleksibel, setelah pelanggan meluncurkan jumlah instans yang ditentukan, setiap instance tambahan yang diluncurkan dibebankan pada tarif per jam yang ditentukan dalam penawaran pribadi penjual. Untuk informasi selengkapnya tentang kontrak multi-tahun dan durasi khusus, lihat Mempersiapkan penawaran pribadi untuk AWS Marketplace produk Anda dan Paket cicilan penawaran pribadi.</p> <p>Setiap Jam dengan Tahunan (termasuk Uji Coba Gratis) — Ini identik dengan model Per Jam dengan opsi Tahunan, kecuali itu termasuk Uji Coba Gratis yang memungkinkan pelanggan untuk menjalankan satu instance dari jenis instans apa pun secara gratis untuk sejumlah hari tertentu yang Anda tentukan. Langganan tahunan dapat dibeli kapan saja, dan digabungkan dengan langganan Uji Coba Gratis.</p> <p>Tahunan dengan Per Jam — Sama seperti model harga Per Jam dengan Tahunan. Pelanggan memiliki opsi untuk membeli penggunaan selama satu tahun di muka untuk satu EC2 contoh Amazon dari satu jenis instance. Anda menetapkan harga untuk setiap jenis instans dan dapat menawarkan penghematan bersih selama harga per jam, tetapi menawarkan penghematan tidak diperlukan. Setiap penggunaan pelanggan di atas jumlah langganan tahunan yang dibeli ditagih dengan tarif per jam yang Anda tetapkan untuk jenis instans tersebut.</p> <p>Durasi Multi-Tahunan dan Kustom dengan Per Jam - Ini hanya tersedia melalui. Mempersiapkan penawaran pribadi untuk AWS Marketplace produk Anda Menggunakan an penawaran pribadi penjual, Anda dapat menentuka</p>

Model penentuan harga	Deskripsi
	<p>n kontrak durasi khusus hingga tiga tahun. Anda dapat meminta pembayaran di muka, atau Anda dapat menawarkan jadwal pembayaran yang fleksibel kepada pelanggan . Anda menetapkan harga untuk setiap jenis instans selama kontrak, dan harga per jam untuk instans tambahan diluncurkan. Jika Anda menawarkan jadwal pembayaran yang fleksibel, Anda juga menetapkan tanggal faktur, jumlah pembayaran, dan jumlah instans untuk setiap jenis instans yang termasuk dalam penawaran. Untuk penawaran pribadi aktif dengan jadwal pembayaran yang fleksibel, setelah jumlah instans yang ditentukan diluncurkan, setiap contoh tambahan yang diluncurkan pelanggan dikenakan biaya pada tarif per jam yang ditentukan dalam penawaran pribadi. Untuk informasi selengkapnya tentang kontrak multi-tahun dan durasi khusus, lihat Mempersiapkan penawaran pribadi untuk AWS Marketplace produk Anda dan Paket cicilan penawaran pribadi.</p> <div data-bbox="646 1052 1507 1266"><p> Note</p><p>Uji Coba Gratis dan Harga Tahunan tidak dapat digabungkan dengan harga Bulanan.</p></div>

Model penentuan harga	Deskripsi
Dibayar bulanan	<p>Bulanan — Perangkat lunak dibayar secara bulanan tetap, terlepas dari jumlah instance yang dijalankan pelanggan . Biaya bulanan dinilai pro saat mendaftar dan setelah pembatalan. Contoh: Pelanggan yang berlangganan selama 1 hari dalam sebulan akan dikenakan biaya untuk 1/30 setiap bulan.</p> <p>Bulanan dengan Per Jam - Biaya Per Jam dan Bulanan diterapkan secara independen. Biaya bulanan dibebankan setiap bulan terlepas dari penggunaan, dan biaya per jam diterapkan hanya berdasarkan penggunaan per jam.</p> <div data-bbox="646 764 1507 982" style="border: 1px solid #add8e6; border-radius: 15px; padding: 10px; margin-top: 10px;"> <p> Note</p> <p>Uji Coba Gratis dan Harga Tahunan tidak dapat digabungkan dengan harga Bulanan.</p> </div>
Harga penggunaan berbayar	<p>Perangkat lunak dibebankan langsung untuk nilai yang Anda berikan bersama dengan salah satu dari empat kategori penggunaan: pengguna, data, bandwidth, atau host. Anda dapat menentukan hingga 24 dimensi untuk produk. Semua biaya masih ditanggung per jam oleh pelanggan.</p> <p>Semua penggunaan dihitung setiap bulan dan ditagih setiap bulan menggunakan mekanisme yang sama dengan AWS Marketplace perangkat lunak yang ada. Harga penggunaan juga disebut sebagai AWS Marketplace Metering Service.</p>
Model harga kontrak	AMI dengan harga kontrak — Produk Single-AMI atau Single-AMI dengan AWS CloudFormation tumpukan yang pembeli membayar biaya di muka.

AWS biaya dan biaya perangkat lunak

Biaya produk berbasis Amazon Machine Image (AMI) terbagi dalam dua kategori:

- **Detail Harga Infrastruktur** — Semua produk berbasis AMI dikenakan biaya AWS infrastruktur terkait tergantung pada layanan dan infrastruktur yang digunakan. Tarif dan biaya ini ditentukan dan dikendalikan oleh AWS, dan dapat bervariasi di antaranya Wilayah AWS. Untuk informasi lebih lanjut, lihat [Amazon EC2 Harga](#).
- **Detail Harga Perangkat Lunak** — Untuk produk Berbayar, penjual menentukan biaya untuk menggunakan perangkat lunak.

Kedua kategori biaya produk ini ditampilkan secara terpisah pada halaman AWS Marketplace detail untuk membantu pembeli memahami potensi biaya penggunaan produk.

Uji coba gratis untuk produk per jam AMI

Produk per jam AMI memenuhi syarat untuk program uji coba gratis opsional. Dalam uji coba Gratis, pelanggan dapat berlangganan produk dan menggunakan satu instance hingga 31 hari tanpa membayar biaya perangkat lunak pada produk. Biaya AWS infrastruktur yang berlaku masih berlaku. Uji coba gratis akan secara otomatis dikonversi ke langganan berbayar setelah kedaluwarsa. Pelanggan akan dikenakan biaya untuk penggunaan tambahan di atas unit gratis yang disediakan. Untuk menawarkan uji coba gratis produk per jam, tentukan durasi periode uji coba dan beri tahu tim [Operasi AWS Marketplace Penjual](#). Masa percobaan bisa 5-31 hari.

Ketika pelanggan berlangganan produk uji coba gratis, mereka menerima pesan email selamat datang yang mencakup jangka waktu uji coba gratis, tanggal kedaluwarsa yang dihitung, dan detail tentang berhenti berlangganan. Pesan email pengingat dikirim tiga hari sebelum tanggal kedaluwarsa.

Jika Anda menawarkan produk uji coba gratis di AWS Marketplace, Anda menyetujui kebijakan pengembalian dana tertentu yang dijelaskan dalam Kebijakan Pengembalian Dana.

Sumber daya tambahan

Untuk informasi selengkapnya tentang harga produk AMI, lihat topik berikut:

- [Harga pengukuran khusus untuk produk AMI dengan AWS Marketplace Metering Service](#)
- [Harga kontrak untuk produk AMI pada AWS Marketplace](#)

Untuk informasi selengkapnya tentang integrasi penagihan, pengukuran, dan lisensi untuk produk berbasis AMI, lihat topik berikut:

tentang harga produk AMI, lihat topik berikut:

- [Mengkonfigurasi pengukuran khusus untuk produk AMI dengan AWS Marketplace Metering Service](#)
- [Mengaitkan lisensi dengan produk berbasis AMI menggunakan AWS License Manager](#)

Harga pengukuran khusus untuk produk AMI dengan AWS Marketplace Metering Service

Dengan AWS Marketplace Metering Service, Anda dapat memodifikasi perangkat lunak Anda untuk mengirim catatan pengukuran ke titik akhir untuk menangkap penggunaan. Anda dapat memilih kategori penggunaan dan menentukan hingga 24 dimensi untuk satu kategori tersebut. Dimensi ini diukur sekali per jam, dikumpulkan, dan dibebankan terhadap rencana harga yang Anda tentukan. Sebagai penjual, Anda harus menentukan dimensi mana yang ingin Anda gunakan. Setelah AMI diterbitkan, Anda tidak akan dapat mengubahnya. Bagian berikut memberikan informasi tentang AWS Marketplace Metering Service.

Topik

- [Menggabungkan konsep layanan pengukuran](#)
- [Harga perangkat lunak Anda](#)
- [Menambahkan produk Anda ke AWS Marketplace](#)
- [Memodifikasi perangkat lunak Anda untuk menggunakan Layanan Pengukuran](#)
- [Penandaan terukur vendor \(Opsional\)](#)
- [Mengkonfigurasi pengukuran kustom](#)
- [Mengkonfigurasi pengukuran khusus untuk produk AMI dengan AWS Marketplace Metering Service](#)

Menggabungkan konsep layanan pengukuran

Sebelum Anda menggunakan AWS Marketplace Metering Service, perhatikan konsep layanan penting berikut ini:

- Kategori Penggunaan — Setiap produk perangkat lunak yang diberi harga melalui penggunaan Layanan Pengukuran dikategorikan menurut satu kategori penggunaan, yang menentukan cara yang tepat untuk membebankan biaya kepada pelanggan. Kategori penggunaan termasuk tetapi tidak terbatas pada:

- **Pengguna** — Satu set izin yang ditentukan yang terkait dengan pengenalan tunggal. Kategori ini sesuai untuk perangkat lunak di mana pengguna pelanggan terhubung ke perangkat lunak secara langsung (misalnya, untuk manajemen hubungan pelanggan atau pelaporan intelijen bisnis).
- **Host** — Setiap server, node, instance, endpoint, atau bagian lain dari sistem komputasi. Kategori ini sesuai untuk perangkat lunak yang memonitor atau memindai banyak instans milik pelanggan (misalnya, pemantauan kinerja atau keamanan).
- **Data** — Penyimpanan atau informasi, diukur dalam MB, GB, atau TB. Kategori ini sesuai untuk perangkat lunak yang mengelola data tersimpan atau memproses data dalam batch.
- **Bandwidth** - Diukur dalam Mbps atau Gbps. Kategori ini sesuai untuk perangkat lunak yang memungkinkan pelanggan untuk menentukan jumlah bandwidth untuk penyediaan.
- **Unit** — Satuan pengukuran; lihat contoh yang dijelaskan selanjutnya.
- **Unit Penggunaan** — Unit penggunaan khusus produk perangkat lunak sesuai dengan kategori penggunaan yang dipilih. Unit penggunaan ini menjelaskan unit yang akan dikenakan biaya perangkat lunak Anda. Contohnya termasuk:
 - **NodesHrs**(sesuai dengan kategori Host)
 - **UserHrs**(sesuai dengan kategori Pengguna)
 - **GBStored**(sesuai dengan kategori Data)
- **Konsumsi** — Produk perangkat lunak yang diberi harga melalui penggunaan Layanan Pengukuran dikenakan biaya untuk konsumsi dalam salah satu dari tiga cara:
 - **Diberikan** - Perangkat lunak ini memungkinkan pelanggan untuk mengkonfigurasi jumlah sumber daya tertentu untuk digunakan (misalnya, jumlah pengguna atau jumlah bandwidth tetap). Setiap jam, pelanggan membayar untuk apa yang telah mereka tetapkan.
 - **Bersamaan** - Perangkat lunak ini memungkinkan sejumlah host yang berbeda atau pengguna untuk terhubung ke perangkat lunak. Setiap jam, pelanggan membayar berdasarkan jumlah host atau pengguna yang mengakses perangkat lunak.
 - **Akumulasi** — Perangkat lunak ini memungkinkan pelanggan untuk menggunakan sejumlah data, baik diproses atau disimpan. Setiap jam, pelanggan membayar jumlah agregat.
- **Harga** — Produk perangkat lunak yang diberi harga melalui penggunaan Layanan Pengukuran harus menentukan harga tunggal atau menentukan hingga 24 dimensi, masing-masing dengan harganya sendiri. Rincian tentang opsi harga meliputi:

- Dimensi tunggal - Ini adalah pilihan harga yang paling mudah. Pelanggan membayar satu harga per unit sumber daya per jam, terlepas dari ukuran atau volume (misalnya, \$0,014 per pengguna per jam, atau \$0,070 per host per jam).
- Multiple dimensions - Opsi harga ini sesuai saat kategori penggunaan yang dipilih bervariasi sepanjang beberapa sumbu. Misalnya, untuk pemantauan host, harga yang berbeda dapat diatur tergantung pada ukuran host. Atau, untuk harga berbasis pengguna, harga yang berbeda dapat ditetapkan berdasarkan tipe pengguna (misalnya, admin, power user, dan read-only user).
- Pengukuran— Semua penggunaan dicatat sebagai acara pengukuran, setiap jam sekali. Perangkat lunak Anda harus dikonfigurasi untuk mengirim dimensi dan jumlah penggunaan yang sesuai ke AWS Marketplace Layanan Pengukuran.
- Alokasi - Opsional, Anda dapat mendistribusikan penggunaan ke alokasi oleh properti yang Anda melacak. Alokasi ini direpresentasikan sebagai tag kepada pembeli. Tag memungkinkan pembeli untuk melihat biaya mereka dibagi menjadi penggunaan dengan tag. Sebagai contoh, jika Anda mengisi oleh pengguna, dan pengguna memiliki properti “Departemen”, Anda dapat membuat alokasi penggunaan dengan tag yang memiliki kunci “Departemen”, dan satu alokasi per nilai. Pendekatan ini tidak mengubah harga, dimensi, atau total penggunaan yang Anda laporkan. Namun, ini memungkinkan pelanggan Anda untuk melihat biaya mereka berdasarkan kategori yang sesuai dengan produk Anda.

Harga perangkat lunak Anda

Saat menetapkan harga perangkat lunak Anda dengan AWS Marketplace Layanan Pengukuran, Anda harus terlebih dahulu memutuskan kategori penggunaan dan bagaimana hal itu akan dikonsumsi. Tia mendukung enam skenario harga yang berbeda. Anda harus memilih hanya salah satu dari ini untuk produk Anda:

- Pengguna yang ditetapkan (per jam)
- Pengguna bersamaan (per jam)
- Host yang ditetapkan (per jam)
- Host serentak (per jam)
- Bandwidth yang ditetapkan (per jam)
- Akumulasi data (per jam)

Selanjutnya, Anda harus memutuskan bagaimana harga kategori penggunaan yang dipilih:

- Harga tunggal
- Beberapa dimensi (hingga 24)

[Menambahkan produk Anda ke AWS Marketplace](#) menjelaskan cara memberikan deskripsi dimensi dan harga yang sesuai dengan pelanggan.

Contoh: Bandwidth yang ditetapkan dengan harga nonlinear

Bayangkan Anda menawarkan perangkat lunak perangkat jaringan. Anda memilih untuk tagihan dengan bandwidth yang ditetapkan. Untuk kategori penggunaan Anda, pilih Bandwidth. Selain pengisian dengan bandwidth, Anda ingin mengisi harga yang berbeda karena pembeli meningkatkan skala. Anda dapat menentukan beberapa dimensi dalam kategori bandwidth. Anda dapat menentukan harga yang berbeda untuk 25 Mbps, 100 Mbps, dan 1 Gbps.

Contoh: host bersamaan dengan beberapa dimensi

Bayangkan Anda menawarkan perangkat lunak yang memantau EC2 instance Amazon lainnya. Anda memilih untuk tagihan dengan jumlah host yang sedang dipantau. Untuk kategori penggunaan Anda, pilih Host. Selain pengisian oleh host, Anda ingin mengisi untuk nilai tambahan untuk memantau host yang lebih besar. Anda dapat menggunakan beberapa dimensi dalam kategori host. Anda dapat menentukan harga yang berbeda untuk mikro, kecil, menengah, besar, x-besar, 2XL, 4XL, dan 8XL contoh. Perangkat lunak Anda bertanggung jawab untuk memetakan setiap host tertentu ke salah satu dimensi yang Anda tentukan. Perangkat lunak Anda bertanggung jawab untuk mengirimkan rekaman pengukuran terpisah untuk setiap dimensi kategori penggunaan Anda jika berlaku.

Menambahkan produk Anda ke AWS Marketplace

Untuk memanfaatkan Layanan Pengukuran, Anda harus membuat produk baru AWS Marketplace untuk dicantumkan. Jika produk Anda sudah ada di AWS Marketplace, Anda harus memutuskan apakah produk AWS Marketplace Metering Service baru akan tersedia selain produk Anda saat ini, atau apakah produk tersebut akan menggantikan produk Anda saat ini sebagai satu-satunya versi yang tersedia untuk pengguna baru. Jika Anda memilih pengganti, produk yang ada akan dihapus dari AWS Marketplace sehingga tidak lagi tersedia untuk pembeli baru. Pelanggan yang sudah ada akan terus memiliki akses ke produk lama mereka dan contoh, tetapi mereka dapat bermigrasi ke produk baru pada kenyamanan mereka. Produk baru harus menggunakan meteran untuk AWS Marketplace Layanan Pengukuran, seperti yang dijelaskan dalam [Memodifikasi perangkat lunak Anda untuk menggunakan Layanan Pengukuran](#).

Setelah Anda memiliki AMI Anda, ikuti proses standar untuk berbagi dan memindai AMI Anda menggunakan alat swalayan. Selain menggunakan templat yang tersedia di portal manajemen, isi formulir beban produk dan unggah untuk memulai proses konsumsi.

Gunakan definisi berikut untuk melengkapi bidang Formulir Pemuatan Produk untuk AWS Marketplace Metering Service. Pada Formulir Beban Produk, bidang ini diberi label sebagai Harga Konsumsi Fleksibel (FCP) untuk membedakannya dari produk dengan harga per jam dan bulanan.

- **Judul** — Jika Anda sudah memiliki produk AWS Marketplace dan Anda menambahkan produk yang sama dengan AWS Marketplace Metering Service, sertakan kategori dan dimensi FCP dalam tanda kurung untuk membedakannya (misalnya, “JUDUL PRODUK (Data)”).
- **Model Harga** — Dari daftar dropdown, pilih Usage.
- **Kategori FCP** — Kategori di mana pelanggan dikenakan biaya untuk produk berbayar dengan komponen harga Penggunaan. Dari daftar pilihan menurun, pilih Pengguna, host, Data, atau Bandwidth.
- **Unit FCP** — Unit pengukuran di mana pelanggan dikenakan biaya untuk produk berbayar dengan komponen harga Penggunaan. Pilihan akan muncul dalam daftar dropdown berdasarkan kategori FCP yang Anda pilih. Tabel berikut menjelaskan unit yang valid untuk setiap kategori.

Kategori	Unit yang valid
Pengguna	UserHrs
host	HostHrs
Data	MB, ID, TB
Bandwidth	Mbps

- **Nama Dimensi FCP** - Nama yang digunakan saat mengirim catatan pengukuran dengan memanggil operasi. `MeterUsage` itu terlihat dalam laporan penagihan. Namun, karena tidak menghadap ke luar, namanya tidak perlu ramah pengguna. Nama tidak boleh lebih dari 15 karakter dan hanya dapat menyertakan karakter alfanumerik dan garis bawah. Setelah Anda menetapkan nama dan membuat produk publik, Anda tidak dapat mengubahnya. Mengubah nama membutuhkan AMI baru.
- **Deskripsi Dimensi FCP** — Pernyataan yang menghadap pelanggan yang menjelaskan dimensi produk. Deskripsi (bisa tidak lebih dari 70 karakter dan harus user-friendly. Contoh deskripsi

meliputi: Administrator per jam dan bandwidth Per Mbps disediakan. Setelah produk diterbitkan, Anda tidak dapat mengubah deskripsi ini.

- Tingkat FCP — Biaya perangkat lunak per unit untuk produk ini. Bidang ini mendukung delapan tempat desimal.

 Catatan:

- Anda tidak perlu mengisi kolom harga per jam dan tahunan.
- Uji coba gratis dan harga tahunan tidak kompatibel.
- Produk yang menggunakan AMI, serta fitur Cluster, dan AWS Resources, tidak dapat menggunakan AWS Marketplace Metering Service.
- Harga, jenis instans, atau Wilayah AWS perubahan akan mengikuti proses yang sama seperti AWS Marketplace produk lainnya.
- Produk dengan AWS Marketplace Metering Service tidak dapat dikonversi ke model harga lain seperti per jam, bulanan, atau Bring Your Own License (BYOL).
- AWS Marketplace merekomendasikan untuk menambahkan informasi kebijakan IAM dalam instruksi atau dokumen penggunaan Anda.
- Anda dapat menyertakan hingga 24 dimensi FCP secara total. Setelah dibuat dan dipublikasikan, Anda tidak dapat mengubah dimensi yang ada, tetapi Anda dapat menambahkan dimensi baru (hingga batas 24).

Jika Anda memiliki pertanyaan, hubungi [AWS Marketplace Operasi Penjual](#) Tim.

Memodifikasi perangkat lunak Anda untuk menggunakan Layanan Pengukuran

Anda perlu memodifikasi perangkat lunak Anda untuk merekam penggunaan pelanggan, mengirim laporan penggunaan per jam ke Layanan Pengukuran, dan menangani mode kegagalan baru. Perangkat lunak ini beroperasi secara independen dari harga, tetapi perangkat lunak perlu mengetahui tentang kategori penggunaan, bagaimana penggunaannya, dan dimensi apa pun.

Mengukur konsumsi

Perangkat lunak Anda harus menentukan berapa banyak kategori penggunaan yang dipilih dan dimensi mana yang telah dikonsumsi pelanggan. Nilai ini akan dikirim, sekali setiap jam, ke Layanan Pengukuran. Dalam semua kasus, diasumsikan bahwa perangkat lunak Anda memiliki kemampuan

untuk mengukur, mencatat, dan membaca konsumsi sumber daya untuk tujuan mengirimkannya setiap jam ke Layanan Pengukuran.

Untuk konsumsi yang ditetapkan, ini biasanya akan dibaca dari konfigurasi perangkat lunak sebagai nilai sampel, tetapi mungkin juga nilai dikonfigurasi maksimum, direkam setiap jam. Untuk konsumsi bersamaan, ini mungkin berupa sampel periodik atau nilai maksimum yang tercatat setiap jam. Untuk akumulasi konsumsi, ini akan menjadi nilai yang terakumulasi setiap jamnya.

Untuk penetapan harga pada beberapa dimensi, beberapa nilai harus diukur dan dikirim ke Layanan Pengukuran, satu per dimensi. Hal ini memerlukan perangkat lunak Anda untuk diprogram atau dikonfigurasi dengan rangkaian dimensi yang diketahui saat Anda memberikan AMI. Kumpulan dimensi tidak dapat berubah setelah produk dibuat.

Untuk setiap skenario harga, tabel berikut menjelaskan cara yang disarankan untuk mengukur konsumsi setiap jam.

Skenario	Bagaimana mengukur
Pengguna yang disediakan	<p>Jumlah pengguna yang ditetapkan saat ini (sampel).</p> <p>-atau-</p> <p>Jumlah maksimum pengguna yang ditetapkan (terlihat jam itu).</p>
Pengguna yang bersamaan	<p>Jumlah pengguna serentak saat ini (sampel).</p> <p>-atau-</p> <p>Jumlah maksimum pengguna bersamaan (terlihat jam itu).</p> <p>-atau-</p> <p>Jumlah total pengguna yang berbeda (terlihat jam itu).</p>
Host yang disediakan	<p>Jumlah host yang ditetapkan saat ini (sampel).</p> <p>-atau-</p>

Skenario	Bagaimana mengukur
	<p>Jumlah maksimum host yang ditetapkan (terlihat jam itu).</p>
Host yang bersamaan	<p>Jumlah host bersamaan saat ini (sampel).</p> <p>-atau-</p> <p>Jumlah maksimum host bersamaan (terlihat jam itu).</p> <p>-atau-</p> <p>Jumlah total host yang berbeda (terlihat jam itu).</p>
Bandwidth yang disediakan	<p>Pengaturan bandwidth yang ditetapkan saat ini (sampel).</p> <p>-atau-</p> <p>Maksimum yang ditetapkan bandwidth (terlihat jam itu).</p>
Mengumpulkan data	<p>GB saat ini data yang disimpan (sampel).</p> <p>-atau-</p> <p>GB maksimum data yang disimpan (dilihat jam itu).</p> <p>-atau-</p> <p>Total GB data yang ditambahkan atau diproses jam tersebut.</p> <p>-atau-</p> <p>Jumlah GB data diproses jam itu.</p>

Penandaan terukur vendor (Opsional)

Penandaan yang diukur vendor membantu Vendor Perangkat Lunak Independen (ISVs) memberi pembeli wawasan yang lebih terperinci tentang penggunaan perangkat lunak mereka dan dapat membantu mereka melakukan alokasi biaya.

Untuk menandai penggunaan perangkat lunak pembeli, Anda perlu menentukan bagaimana biaya dialokasikan. Pertama tanyakan kepada pembeli Anda apa yang ingin mereka lihat dalam alokasi biaya mereka. Kemudian Anda dapat membagi penggunaan di seluruh properti yang Anda lacak untuk akun pembeli. Contoh properti termasuk Account ID, Business Unit Cost Centers, dan metadata relevan lainnya untuk produk Anda. Properti ini diekspos ke pembeli sebagai tag. Dengan menggunakan tag, pembeli dapat melihat biayanya dibagi menjadi penggunaan berdasarkan nilai tag di Konsol AWS Penagihan (<https://console.aws.amazon.com/costmanagement/>). Penandaan yang diukur vendor tidak mengubah harga, dimensi, atau total penggunaan yang Anda laporkan. Hal ini memungkinkan pelanggan Anda untuk melihat biaya mereka berdasarkan kategori yang sesuai dengan produk Anda.

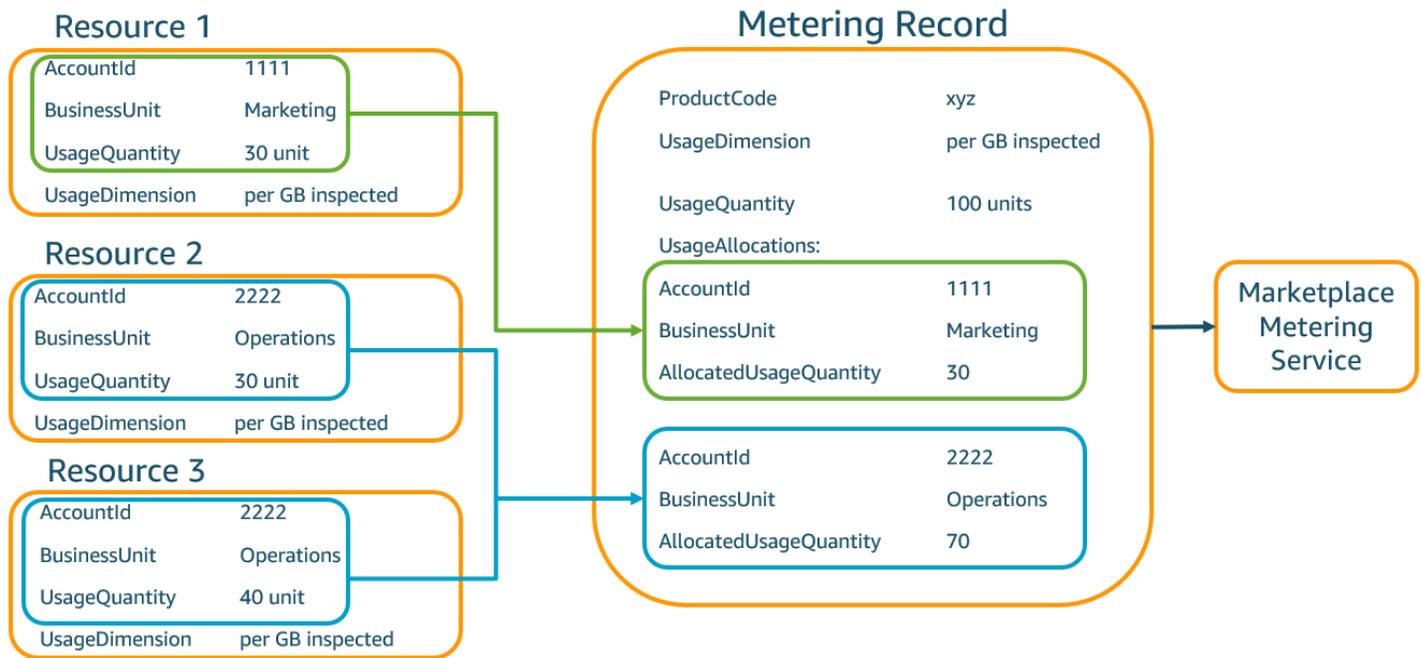
Dalam kasus penggunaan umum, pembeli berlangganan produk Anda dengan satu Akun AWS. Pembeli juga memiliki banyak pengguna yang terkait dengan langganan produk yang sama. Anda dapat membuat alokasi penggunaan dengan tag yang memiliki kunci Account ID, dan kemudian mengalokasikan penggunaan untuk setiap pengguna. Dalam hal ini, pembeli dapat mengaktifkan Account ID tag di konsol Billing and Cost Management mereka dan menganalisis penggunaan pengguna individu.

Pengalaman penjual

Penjual dapat menggabungkan catatan pengukuran untuk sumber daya dengan kumpulan tag yang sama alih-alih menggabungkan penggunaan untuk semua sumber daya. Misalnya, penjual dapat membuat catatan pengukuran yang mencakup ember yang berbeda. UsageAllocations Setiap bucket mewakili UsageQuantity satu set tag, seperti AccountId dan BusinessUnit.

Dalam diagram berikut, Resource 1 memiliki satu set AccountId dan BusinessUnit tag yang unik, dan muncul di Rekaman Pengukuran sebagai entri tunggal.

Resource 2 dan Resource 3 keduanya memiliki AccountId tag yang sama 2222, dan BusinessUnit tag yang sama, Operations. Akibatnya, mereka digabungkan menjadi satu UsageAllocations entri dalam Rekaman Pengukuran.



Penjual juga dapat menggabungkan sumber daya tanpa tag menjadi satu UsageAllocation dengan jumlah penggunaan yang dialokasikan dan mengirimkannya sebagai salah satu entri di UsageAllocations

Batasan meliputi:

- Jumlah tag — 5
- Ukuran UsageAllocations (kardinalitas) — 2.500

Validasi meliputi:

- Karakter diperbolehkan untuk kunci tag dan nilai — a-zA-Z 0-9+ -= . _:\ /@
- Tag maksimum di seluruh UsageAllocation daftar - 5
- Dua tidak UsageAllocations dapat memiliki tag yang sama (yaitu, kombinasi yang sama dari kunci tag dan nilai). Jika itu masalahnya, mereka harus menggunakan hal yang samaUsageAllocation.
- Jumlah AllocatedUsageQuantity UsageAllocation harus sama denganUsageQuantity, yang merupakan penggunaan agregat.

Pengalaman pembeli

Tabel berikut menunjukkan contoh pengalaman pembeli setelah pembeli mengaktifkan tag `AccountId` dan `BusinessUnit` vendor.

Dalam contoh ini, pembeli dapat melihat penggunaan yang dialokasikan dalam Laporan Penggunaan Biaya mereka. Tag yang diukur vendor menggunakan awalan. “`aws:marketplace:isv`” Pembeli dapat mengaktifkannya di Billing and Cost Management, di bawah Tag Alokasi Biaya AWS, tag alokasi biaya yang dihasilkan.

Baris pertama dan terakhir dari Laporan Penggunaan Biaya relevan dengan apa yang dikirim Penjual ke Layanan Pengukuran (seperti yang ditunjukkan dalam [Pengalaman penjual](#) contoh).

Laporan Penggunaan Biaya (Sederhana)

ProductCode	Pembeli	UsageDimension	UsageQuantity	<code>aws:marketplace:isv:AccountId</code>	<code>aws:marketplace:isv:BusinessUnit</code>
xyz	111122223333	Jaringan: per (GB) diperiksa	70	2222	Operasi
xyz	111122223333	Jaringan: per (GB) diperiksa	30	3333	Keuangan
xyz	111122223333	Jaringan: per (GB) diperiksa	20	4444	IA
xyz	111122223333	Jaringan: per (GB) diperiksa	20	5555	Pemasaran
xyz	111122223333	Jaringan: per (GB) diperiksa	30	1111	Pemasaran

Untuk contoh kode, lihat [MeterUsage dengan penandaan alokasi penggunaan \(Opsional\)](#)

Mengkonfigurasi pengukuran kustom

Untuk informasi selengkapnya tentang bekerja dengan AWS Marketplace Metering Service, lihat [Mengkonfigurasi pengukuran khusus untuk produk AMI dengan AWS Marketplace Metering Service](#).

Mengkonfigurasi pengukuran khusus untuk produk AMI dengan AWS Marketplace Metering Service

Note

Untuk produk berbasis AMI dengan harga pengukuran khusus, perangkat lunak Anda harus memanggil [MeterUsage API](#) menggunakan AWS kredensi sementara peran IAM untuk Amazon Elastic Compute Cloud yang dilampirkan ke instans Amazon. EC2 Menggunakan kunci akses jangka panjang tidak didukung.

AWS Marketplace Metering Service adalah fitur penetapan harga dan pengukuran yang dapat Anda gunakan untuk menagih langsung perangkat lunak berdasarkan kategori penggunaan. Ada lima kategori penggunaan: pengguna, data, bandwidth, host, atau unit. Anda dapat menggunakan Layanan Pengukuran dengan produk berbasis Amazon Machine Image (AMI), berbasis kontainer, dan perangkat lunak sebagai layanan (SaaS). Bagian berikut memberikan informasi selengkapnya tentang cara mengonfigurasi pengukuran khusus dengan AWS Marketplace Metering Service.

AWS Marketplace Layanan Pengukuran memungkinkan beberapa skenario baru. Misalnya, jika monitor perangkat lunak Anda host, Anda dapat mengisi daya untuk setiap host dipantau. Anda dapat memiliki harga yang berbeda berdasarkan ukuran host, dan biaya untuk jumlah host bersamaan dipantau setiap jam. Demikian pula, jika perangkat lunak Anda memungkinkan banyak pengguna di seluruh organisasi untuk masuk, Anda dapat mengisi daya dengan jumlah pengguna. Setiap jam, pelanggan dikenakan biaya untuk jumlah total pengguna yang ditetapkan.

Untuk informasi selengkapnya, lihat [AWS Marketplace Layanan Pengukuran API Referensi](#).

Untuk informasi selengkapnya tentang mengintegrasikan AWS Marketplace Metering Service API untuk produk berbasis AMI dengan harga pengukuran khusus, lihat Daftar [produk AMI dengan harga laboratorium unit kustom bengkel penjual](#). AWS Marketplace

Topik

- [Persyaratan](#)
- [Hubungi AWS Marketplace Layanan Pengukuran](#)
- [Penanganan kegagalan](#)
- [Batasan](#)
- [Contoh kode](#)

Persyaratan

Semua perangkat lunak berbasis AMI yang menggunakan Layanan Metering harus memenuhi persyaratan berikut:

- Perangkat lunak Anda harus diluncurkan AWS Marketplace melalui Amazon Machine Image (AMI).
- Jika Anda memiliki produk yang sudah ada AWS Marketplace, Anda harus mengirimkan AMI baru dan membuat produk baru untuk mengaktifkan fitur ini.
- Semua perangkat lunak harus disediakan dengan peran AWS Identity and Access Management (IAM). Pelanggan akhir harus menambahkan peran IAM ke instans Amazon Elastic Compute Cloud EC2 (Amazon) yang disediakan pengguna dengan perangkat lunak tersebut. Penggunaan peran IAM bersifat opsional saat Anda menggunakan perangkat lunak. AWS Marketplace Ini diperlukan saat Anda menerapkan perangkat lunak AWS Marketplace Metering Service.
- Perangkat lunak Anda harus dapat menentukan konsumsi dalam beberapa cara.

Hubungi AWS Marketplace Layanan Pengukuran

Perangkat lunak Anda harus menghubungi Layanan Pengukuran per jam dan mencatat nilai konsumsi untuk jam tersebut.

Ketika perangkat lunak Anda dimulai, itu harus merekam minute-of-the-hour di mana ia dimulai. Ini disebut sebagai menit awal. Setiap jam pada menit awal, perangkat lunak Anda harus menentukan nilai konsumsi untuk jam itu dan menghubungi Layanan Pengukuran. Untuk informasi tentang cara mendapatkan nilai ini, lihat [Memodifikasi perangkat lunak Anda untuk menggunakan Layanan Pengukuran](#).

Untuk bangun setiap jam di menit awal, perangkat lunak Anda harus menggunakan salah satu pendekatan berikut:

- Sebuah thread dalam perangkat lunak Anda.

- Proses daemon yang dimulai dengan instans atau perangkat lunak.
- Pekerjaan cron yang dikonfigurasi selama aplikasi startup.

 Note

Perangkat lunak Anda harus memanggil AWS Marketplace Layanan Pengukuran menggunakan IAM role dikonfigurasi pada contoh pelanggan dan menentukan dimensi konsumsi dan jumlah.

Perangkat lunak Anda dapat menggunakan AWS SDK untuk memanggil AWS Marketplace Metering Service, mirip dengan contoh implementasi berikut:

1. Gunakan profil contoh untuk membuat klien layanan. Ini membutuhkan peran yang dikonfigurasi untuk EC2 instance. Kredensial peran disegarkan oleh SDK secara otomatis.
2. Setiap jam, baca konfigurasi perangkat lunak Anda dan negara untuk menentukan nilai konsumsi untuk jam tersebut. Ini mungkin termasuk mengumpulkan value-per-dimension.
3. Panggil `meterUsage` pada klien SDK dengan parameter berikut (panggilan tambahan untuk setiap dimensi yang memiliki penggunaan):
 - `timestamp`— Stempel waktu jam yang direkam (dalam UTC).
 - `productCode`— Kode produk yang ditetapkan untuk perangkat lunak.
 - `dimension`— Dimensi (atau dimensi) yang ditetapkan untuk perangkat lunak.
 - `quantity`— Nilai konsumsi untuk satu jam.
 - `allocations`— (Opsional) Anda dapat memberikan alokasi untuk penggunaan di seluruh properti yang Anda lacak. Alokasi ini harus menambahkan hingga total konsumsi dalam catatan. Bagi pembeli, ini ditampilkan sebagai tag alokasi biaya potensial di alat penagihan mereka (seperti AWS Manajemen Penagihan dan Biaya konsol). Pembeli harus mengaktifkan tag di akun mereka untuk melacak biaya mereka menggunakan tag ini.

Selain itu, perangkat lunak Anda harus menghubungi endpoint AWS Marketplace Layanan Pengukuran di wilayah. Produk Anda harus memiliki pengaturan titik akhir Regional yang benar, sehingga `us-east-1` mengirimkan catatan ke `us-east-1` titik akhir, dan `us-west-2` mengirimkan catatan ke titik akhir. `us-west-2` Melakukan panggilan di wilayah memberi pembeli pengalaman yang lebih stabil dan mencegah situasi di mana ketersediaan Wilayah yang tidak terkait dapat memengaruhi perangkat lunak yang berjalan di Wilayah lain.

Saat Anda mengirim catatan pengukuran ke layanan, Anda harus terhubung ke AWS Marketplace Layanan Pengukuran di Wilayah Anda. Gunakan metode `getCurrentRegion()` helper untuk menentukan Region di mana EC2 instance berjalan, dan kemudian meneruskan informasi Region ini ke `MeteringServiceClient` konstruktor. Jika Anda tidak menentukan Wilayah AWS dalam konstruktor SDK, `us-east-1` Region default akan digunakan. Jika aplikasi Anda mencoba untuk membuat Cross-region panggilan ke layanan, panggilan ditolak. Untuk informasi selengkapnya, lihat [Menentukan Wilayah Saat Ini Aplikasi](#) dan [getCurrentRegion\(\)](#).

Penanganan kegagalan

Produk Anda harus mengirimkan catatan pengukuran ke layanan, titik akhir internet publik, sehingga penggunaan dapat diambil dan ditagih. Karena mungkin bagi pelanggan untuk memodifikasi pengaturan jaringan dengan cara yang mencegah catatan pengukuran Anda dikirim, produk Anda harus memperhitungkannya dengan memilih mode kegagalan.

Note

Beberapa kegagalan pengukuran mungkin merupakan masalah sementara dalam menghubungkan ke AWS Marketplace Metering Service AWS Marketplace sangat merekomendasikan menerapkan percobaan ulang hingga 30 menit, dengan mundur eksponensial, untuk menghindari pemadaman jangka pendek atau masalah jaringan.

Biasanya, perangkat lunak dapat gagal terbuka (memberikan pesan peringatan tetapi mempertahankan fungsionalitas penuh) atau gagal ditutup (menonaktifkan semua fungsi dalam aplikasi sampai sambungan telah dibuat kembali). Anda dapat memilih untuk gagal membuka, menutup, atau sesuatu yang spesifik untuk aplikasi Anda. Kami sangat menyarankan agar Anda menahan diri dari gagal ditutup setelah kurang dari dua jam pengukuran kegagalan.

Sebagai contoh gagal sebagian terbuka, Anda dapat terus mengizinkan akses ke perangkat lunak tetapi tidak memungkinkan pembeli untuk memodifikasi pengaturan perangkat lunak. Atau, pembeli masih dapat mengakses perangkat lunak tetapi tidak akan dapat membuat pengguna tambahan. Perangkat lunak Anda bertanggung jawab untuk menentukan dan menegakkan mode kegagalan ini. Mode kegagalan perangkat lunak Anda harus disertakan saat AMI Anda dikirimkan, dan tidak dapat diubah nanti.

Batasan

Ingatlah keterbatasan ini saat merancang dan mengirimkan perangkat lunak berkemampuan Layanan Pengukuran-enabled Anda:

- Persyaratan peran dan gateway internet IAM untuk pelanggan Anda— Pelanggan Anda harus memiliki gateway internet dan harus meluncurkan perangkat lunak Anda dengan IAM role dengan izin tertentu. Untuk informasi selengkapnya, lihat [AWS Marketplace izin API pengukuran dan hak](#). Perangkat lunak Anda tidak dapat terhubung ke Layanan Pengukuran jika kedua kondisi ini tidak terpenuhi.
- Ketidakmampuan untuk menambahkan kategori penggunaan baru atau mengubah ke produk Layanan Pengukuran yang ada — Ketika pelanggan berlangganan produk perangkat lunak Anda, mereka menyetujui syarat dan ketentuan. Mengubah Kategori penggunaandalam produk dengan Layanan Pengukuran memerlukan produk baru dan langganan baru.
- Ketidakmampuan untuk mengubah dimensi ke produk Layanan Pengukuran yang ada — Ketika pelanggan berlangganan produk perangkat lunak Anda, mereka menyetujui syarat dan ketentuan. Mengubah dimensi dalam produk dengan Layanan Pengukuran memerlukan produk baru dan langganan baru. Anda dapat menambahkan dimensi baru untuk produk yang ada, hingga batas 24.
- Kurangnya uji coba gratis dan langganan tahunan — Produk Metering Service tidak mendukung uji coba gratis dan langganan tahunan saat diluncurkan.
- Pertimbangan penggunaan multi-instans atau berbasis cluster— Beberapa perangkat lunak digunakan sebagai bagian dari deployment multi-instance. Saat Anda merancang perangkat lunak Anda, pertimbangkan bagaimana dan di mana konsumsi diukur dan di mana catatan pengukuran dipancarkan.

Contoh kode

Contoh kode berikut disediakan untuk membantu Anda mengintegrasikan produk AMI Anda dengan yang AWS Marketplace APIs diperlukan untuk menerbitkan dan memelihara produk Anda.

MeterUsage dengan penandaan alokasi penggunaan (Opsional)

Contoh kode berikut relevan untuk produk AMI dengan model harga konsumsi. Contoh Python mengirimkan catatan pengukuran dengan tag alokasi penggunaan yang sesuai untuk membebaskan biaya AWS Marketplace kepada pelanggan Anda. pay-as-you-go

```
# NOTE: Your application will need to aggregate usage for the
#         customer for the hour and set the quantity as seen below.
```

```
# AWS Marketplace can only accept records for up to an hour in the past.
#
# productCode is supplied after the AWS Marketplace Ops team has
# published the product to limited

# Import AWS Python SDK
import boto3
import time

usageRecord = [
    {
        "AllocatedUsageQuantity": 2,
        "Tags":
            [
                { "Key": "BusinessUnit", "Value": "IT" },
                { "Key": "AccountId", "Value": "123456789" },
            ]
    },
    {
        "AllocatedUsageQuantity": 1,
        "Tags":
            [
                { "Key": "BusinessUnit", "Value": "Finance" },
                { "Key": "AccountId", "Value": "987654321" },
            ]
    }
]

marketplaceClient = boto3.client("meteringmarketplace")

response = marketplaceClient.meter_usage(
    ProductCode="testProduct",
    Timestamp=int(time.time()),
    UsageDimension="Dimension1",
    UsageQuantity=3,
    DryRun=False,
    UsageAllocations=usageRecord
)
```

Untuk informasi selengkapnya `MeterUsage`, lihat [MeterUsage](#) di Referensi AWS Marketplace API Layanan Pengukuran.

Contoh tanggapan

```
{ "MeteringRecordId": "string" }
```

Harga kontrak untuk produk AMI pada AWS Marketplace

Harga kontrak untuk produk berbasis Amazon Machine Image (AMI) berarti bahwa pembeli membayar biaya di muka untuk satu produk AMI atau AMI tunggal dengan AWS CloudFormation tumpukan. Untuk produk berbasis AMI dengan harga kontrak, AWS Marketplace menagih pelanggan Anda di muka atau dengan jadwal pembayaran yang Anda tentukan, berdasarkan kontrak antara Anda dan pelanggan Anda. Setelah itu, mereka berhak menggunakan sumber daya tersebut. Topik ini memberikan informasi lebih lanjut tentang harga kontrak.

Untuk menetapkan harga Anda, pilih satu atau lebih durasi kontrak yang Anda tawarkan kepada pelanggan. Anda dapat memasukkan harga yang berbeda untuk setiap durasi kontrak. Pilihan Anda adalah durasi 1 bulan, 12 bulan, 24 bulan, dan 36 bulan. Untuk penawaran pribadi, Anda dapat menentukan durasi khusus dalam beberapa bulan (hingga 60 bulan).

Note

Uji coba gratis tidak tersedia untuk produk AMI dengan harga kontrak.

Pilih kategori yang paling menggambarkan harga produk Anda. Kategori harga muncul untuk pelanggan di situs AWS Marketplace web. Anda dapat memilih dari Bandwidth (GB/s, MB/s), Data (GB, MB, TB), Host, Requests, Tiers, atau Users. Jika tidak ada kategori standar yang sesuai dengan kebutuhan Anda, Anda dapat memilih kategori Unit yang lebih umum.

Penawaran ini memungkinkan hingga 24 dimensi untuk ditambahkan ke dalamnya. Setiap dimensi membutuhkan data berikut:

- **Kategori Kontrak** — Untuk produk kontrak tanpa harga berbasis konsumsi, Anda dapat memilih kategori yang paling mirip dengan kategori dimensi dalam kontrak atau memilih Unit jika tidak ada nilai yang menyerupai unit untuk dimensi dalam kontrak
- **Unit Kontrak** — Pilih salah satu nilai yang tersedia untuk unit yang sangat cocok dengan dimensi Anda berdasarkan Kategori yang dipilih.
- **Dimensi Kontrak Memungkinkan Beberapa Pembelian** — Bidang ini digunakan untuk menunjukkan apakah penawaran adalah penawaran harga berjenjang atau penawaran yang tidak berjenjang:

Penawaran berjenjang — Memungkinkan pembeli untuk berlangganan hanya salah satu dimensi yang tersedia dalam penawaran. Dimensi dalam penawaran berjenjang tidak memiliki konsep kuantitas. Menandatangani kontrak dengan dimensi tertentu pada dasarnya menunjukkan bahwa pembeli telah memilih fitur spesifik yang ditunjukkan oleh dimensi itu.

Penawaran tidak berjenjang — Memungkinkan pelanggan untuk mendapatkan lebih dari satu dimensi sebagai bagian dari kontrak dan memungkinkan mereka untuk mendapatkan beberapa unit dari setiap dimensi tersebut.

Menetapkan nilai `true` untuk bidang ini menunjukkan bahwa penawaran tersebut adalah penawaran yang tidak berjenjang. Menetapkan nilai `false` untuk bidang ini menunjukkan bahwa penawaran tersebut adalah penawaran berjenjang.

Saat menggunakan Product Load Form (PLF) untuk membuat kontrak untuk produk AMI Anda, Anda harus menentukan bidang berikut untuk dimensi harga Anda:

- **Contracts DimensionX API Name** — Nama yang akan muncul dalam lisensi yang dihasilkan di akun pembeli. AWS License Manager Nama ini juga digunakan sebagai nilai untuk `Name in Entitlement` dalam panggilan `CheckoutLicense API`.
- **Contracts DimensionX Display Name** — Nama dimensi yang menghadap pelanggan yang akan ditampilkan pada detail produk dan halaman pengadaan situs web. AWS Marketplace Buat nama yang ramah pengguna. Panjang maksimum nama adalah 24 karakter. Setelah daftar bersifat publik, nilai tidak Name dapat diubah.
- **Deskripsi DimensionX Kontrak** — Deskripsi dimensi yang dihadapi pelanggan yang memberikan informasi tambahan tentang dimensi produk, seperti kemampuan yang disediakan dimensi tertentu. Panjang maksimum untuk deskripsi adalah 70 karakter.
- **Contracts DimensionX Quantity** — Ini digunakan untuk menghitung prorasi dalam kasus amandemen perjanjian terhadap suatu produk. Nilai bidang ini harus diatur ke 1 untuk semua penawaran kontrak. Seharusnya tidak diedit.
- **Kontrak DimensionX Tarif 1 Bulan** — Tarif kontrak yang akan dikenakan untuk satu bulan hak terhadap dimensi ini. Untuk penawaran non-tier, tarif ini dibebankan untuk setiap unit dimensi yang diperoleh. Bidang ini mendukung tiga angka desimal.
- **Kontrak DimensionX 12-Bulan Tarif** — Tarif kontrak yang akan dibebankan selama 12 bulan hak terhadap dimensi tersebut. Untuk penawaran non-tier, tarif ini dibebankan untuk setiap unit dimensi yang diperoleh. Bidang ini mendukung tiga angka desimal.

- Kontrak DimensionX Tarif 24 Bulan — Tarif kontrak yang akan dibebankan selama 24 bulan hak terhadap dimensi tersebut. Untuk penawaran non-tier, tarif ini dibebankan untuk setiap unit dimensi yang diperoleh. Bidang ini mendukung tiga angka desimal.
- Kontrak DimensionX 36-Month Rate — Tarif kontrak yang akan dibebankan selama 36 bulan hak terhadap dimensi tersebut. Untuk penawaran non-tier, tarif ini dibebankan untuk setiap unit dimensi yang diperoleh. Bidang ini mendukung tiga angka desimal.

Contoh: Aplikasi penyimpanan data

	Harga 1 bulan	Harga 12 bulan	Harga 24 bulan	Harga 36 bulan
Data tidak terenkripsi (GB)	\$1,50/GB	\$16,00/GB	\$30.00/GB	\$60.00/GB
Data terenkripsi (GB)	\$1,55/GB	\$16,60/GB	\$31.20/GB	\$61.20/GB

Contoh: Produk pemantauan log

	Harga 1 bulan	Harga 12 bulan	Harga 24 bulan	Harga 36 bulan
Dasar (10 host dipantau, 5 kontainer dipantau)	\$100	\$1000	\$2000	\$4000
Standar (20 host dipantau, 10 kontainer dipantau)	\$200	\$2000	\$4000	\$8000
Pro (40 host dipantau, 20 kontainer dipantau)	\$400	\$4000	\$8000	\$16.000
Host tambahan dipantau per jam	\$10	\$100	\$200	\$400

	Harga 1 bulan	Harga 12 bulan	Harga 24 bulan	Harga 36 bulan
Kontainer tambahan dipantau per jam	\$10	\$100	\$200	\$400

Note

Harga bisa untuk jangka waktu berikut: 1 bulan, 12 bulan, 24 bulan, atau 36 bulan. Anda dapat memilih untuk menawarkan satu atau lebih opsi ini untuk produk Anda. Durasi harus sama di setiap dimensi.

Example

Misalnya, dalam kasus di mana Anda memiliki `ReadOnlyUsers` dan `AdminUsers` dimensi, jika Anda menawarkan harga tahunan untuk `ReadOnlyUsers`, Anda harus menawarkan harga tahunan juga. `AdminUsers`

Perpanjangan otomatis

Ketika pelanggan membeli produk Anda melalui AWS Marketplace menggunakan kontrak AMI, mereka dapat setuju untuk memperbarui ketentuan kontrak secara otomatis. Pelanggan terus membayar hak setiap bulan atau selama 1, 2, atau 3 tahun.

Pelanggan dapat mengubah pengaturan perpanjangan mereka kapan saja. Untuk informasi selengkapnya, lihat [Memodifikasi kontrak yang ada](#) di Panduan AWS Marketplace Pembeli.

Mengaitkan lisensi dengan produk berbasis AMI menggunakan AWS License Manager

Untuk produk berbasis Amazon Machine Image (AMI) dengan harga kontrak, Anda dapat menggunakan AWS License Manager untuk mengaitkan lisensi dengan produk Anda. AWS License Manager adalah alat manajemen lisensi yang memungkinkan aplikasi Anda untuk melacak dan memperbarui lisensi (juga dikenal sebagai hak) yang telah dibeli oleh pelanggan. Setelah integrasi selesai, Anda dapat mempublikasikan daftar produk Anda AWS Marketplace. Bagian berikut memberikan informasi lebih lanjut tentang penggunaan AWS License Manager untuk mengaitkan lisensi dengan produk berbasis AMI Anda.

Untuk informasi selengkapnya AWS License Manager, lihat [Panduan AWS License Manager Pengguna](#) dan [AWS License Manager](#) bagian Referensi AWS CLI Perintah.

Untuk informasi lebih lanjut tentang integrasi AWS License Manager dengan produk berbasis AMI dengan harga kontrak, lihat Daftar [produk AMI dengan harga berdasarkan lab pembayaran di muka bengkel penjual](#).AWS Marketplace

Note

- Pelanggan tidak dapat meluncurkan instans baru AMI setelah masa kontrak berakhir. Namun, selama durasi kontrak, mereka dapat meluncurkan sejumlah instance. Lisensi ini tidak dikunci simpul atau terikat pada instance tertentu.
- Pembuatan Penawaran Pribadi — Penjual dapat menghasilkan penawaran pribadi untuk produk menggunakan alat pembuatan penawaran Pribadi di Portal Manajemen AWS Marketplace.
- Pelaporan — Anda dapat mengatur umpan data dengan menyiapkan bucket Amazon S3 di bagian Laporan di bagian. Portal Manajemen AWS Marketplace Untuk informasi lebih lanjut, lihat [Laporan penjual, umpan data, dan dasbor di AWS Marketplace](#).

Topik

- [Model lisensi](#)
- [Alur kerja integrasi](#)
- [Prasyarat integrasi License Manager](#)
- [Mengintegrasikan produk berbasis AMI dengan AWS License Manager](#)
- [Perpanjangan dan peningkatan lisensi](#)

Model lisensi

AWS Marketplace integrasi dengan AWS License Manager mendukung dua model lisensi:

- [Model lisensi yang dapat dikonfigurasi](#)
- [Model lisensi berjenjang](#)

Model lisensi yang dapat dikonfigurasi

Model lisensi yang dapat dikonfigurasi (juga dikenal sebagai model lisensi yang dapat diukur) memberikan hak kepada pembeli untuk sejumlah sumber daya tertentu setelah pembeli memperoleh lisensi.

Anda menetapkan dimensi harga dan harga per unit. Kemudian, pembeli dapat memilih jumlah sumber daya yang ingin mereka beli.

Example dimensi harga dan harga per unit

Anda dapat menetapkan dimensi harga (seperti cadangan data) dan harga per unit (seperti \$30/unit)

Pembeli dapat memilih untuk membeli 5, 10, atau 20 unit.

Produk Anda melacak dan penggunaan meter untuk mengukur jumlah sumber daya yang dikonsumsi.

Dengan model konfigurasi, hak dihitung dalam salah satu dari dua cara:

- [Lisensi drawdown](#)
- [Lisensi mengambang](#)

Lisensi drawdown

Lisensi diambil dari kumpulan jumlah lisensi yang diizinkan saat digunakan. Hak tersebut diperiksa secara permanen dan tidak dapat dikembalikan ke kumpulan lisensi.

Example Memproses data dalam jumlah terbatas

Seorang pengguna berhak untuk memproses 500 GB data. Ketika mereka terus memproses data, kuantitas diambil dari kumpulan 500 GB hingga semua lisensi 500 GB dikonsumsi.

Untuk lisensi drawdown, Anda dapat menggunakan operasi `CheckoutLicense` API untuk memeriksa unit lisensi yang digunakan.

Example cadangan ke Amazon S3 untuk sejumlah unit/tahun

Anda memiliki produk penyimpanan yang memungkinkan pencadangan ke Amazon Simple Storage Service hingga 1024 unit untuk data selama satu tahun. Aplikasi Anda dapat diluncurkan dengan menggunakan beberapa EC2 instans Amazon. Aplikasi Anda memiliki mekanisme untuk melacak dan

mengumpulkan data. Perangkat lunak Anda memanggil operasi CheckoutLicense API dengan ID Produk pada setiap cadangan atau pada interval tetap untuk memperbarui jumlah yang dikonsumsi.

Dalam contoh ini, perangkat lunak Anda memanggil CheckoutLicense untuk memeriksa 10 unit data. Ketika total kapasitas mencapai batas cadangan yang telah dibeli pelanggan, panggilan API gagal.

Permintaan

```
linux-machine ~]$ aws license-manager checkout-license\  
--product-sku "2205b290-19e6-4c76-9eea-377d6bf71a47" \  
--checkout-type "PERPETUAL" \  
--key-fingerprint "aws:294406891311:AWS/Marketplace:issuer-fingerprint" \  
--entitlements "Name=DataConsumption, Value=10, Unit=Count" \  
--client-token "AKIAIOSFODNN7EXAMPLE"
```

Respons

```
{  
  "CheckoutType": "PERPETUAL",  
  "EntitlementsAllowed": [  
    {  
      "Name": "DataConsumption",  
      "Count": 10,  
      "Units": "Count",  
      "Value": "Enabled"  
    }  
  ],  
  "Expiration": "2021-04-22T19:02:36",  
  "IssuedAt": "2021-04-22T18:02:36",  
  "LicenseArn": "arn:aws:license-manager::294406891311:license:l-16bf01b...",  
  "LicenseConsumptionToken": "AKIAIOSFODNN7EXAMPLE"  
}
```

Lisensi mengambang

Lisensi dikembalikan ke kumpulan jumlah lisensi yang diizinkan setelah digunakan.

Example jumlah pengguna dari batas atas tetap

Seorang pengguna berhak atas 500 pengguna simultan pada aplikasi. Saat pengguna masuk dan keluar, pengguna ditarik dan dikembalikan ke kumpulan 500 pengguna. Namun, aplikasi tidak dapat menarik lebih dari 500 pengguna dari pool karena 500 pengguna simultan adalah batas atas tetap.

Untuk lisensi mengambang, Anda dapat menggunakan operasi `CheckInLicense` API untuk mengembalikan unit lisensi ke kumpulan hak.

Example jumlah pengguna bersamaan selama satu tahun

Harga produk Anda berdasarkan jumlah pengguna bersamaan. Pelanggan membeli lisensi untuk 10 pengguna selama satu tahun. Pelanggan meluncurkan perangkat lunak dengan memberikan izin AWS Identity and Access Management (IAM). Ketika pengguna log in, aplikasi Anda memanggil operasi `CheckoutLicense` API untuk mengurangi kuantitas sebesar 1. Saat pengguna log out, aplikasi mengembalikan lisensi tersebut ke pool dengan memanggil operasi `CheckInLicense` API. Jika Anda tidak menelepon `CheckInLicense`, unit lisensi akan secara otomatis diperiksa setelah 1 jam.

Note

Dalam Permintaan berikut, `key-fingerprint` bukan nilai placeholder tetapi nilai sebenarnya dari sidik jari yang dengannya semua lisensi akan dipublikasikan.

Permintaan

```
linux-machine ~]$ aws license-manager checkout-license \
--product-sku "2205b290-19e6-4c76-9eea-377d6bf71a47" \
--checkout-type "PROVISIONAL" \
--key-fingerprint "aws:294406891311:AWS/Marketplace:issuer-fingerprint" \
--entitlements "Name=ReadOnlyUSers, Value=10, Unit=Count" \
--client-token "AKIAIOSFODNN7EXAMPLE"
```

Respons

```
{
  "CheckoutType": "PROVISIONAL",
  "EntitlementsAllowed": [
    {
```

```
    "Name": "ReadOnlyUsers",
    "Count": 10,
    "Units": "Count",
    "Value": "Enabled"
  }
},
"Expiration": "2021-04-22T19:02:36",
"IssuedAt": "2021-04-22T18:02:36",
"LicenseArn": "arn:aws:license-manager::294406891311:license:l-16bf01b...",
"LicenseConsumptionToken": "AKIAIOSFODNN7EXAMPLE"
}
```

Model lisensi berjenjang

Model lisensi berjenjang memberikan hak kepada pembeli ke tingkat tertentu, atau tingkat, fitur aplikasi setelah pembeli memperoleh lisensi.

Anda membuat tingkatan untuk produk Anda, seperti Basic, Intermediate, dan Premium. Pembeli kemudian memilih salah satu tingkatan yang telah ditentukan.

Aplikasi tidak perlu melacak atau mengukur penggunaan aplikasi.

Dengan model lisensi berjenjang, hak tidak dihitung melainkan menandakan tingkat layanan yang diperoleh oleh pelanggan.

Jika Anda ingin menawarkan fitur yang dibundel bersama-sama, kami sarankan menggunakan model lisensi berjenjang.

Example Tingkat Dasar, Menengah, dan Premium

Pelanggan dapat menandatangani kontrak untuk salah satu dari tiga kemungkinan tingkatan perangkat lunak: Dasar, Menengah, atau Premium. Masing-masing tingkatan ini memiliki harga sendiri. Perangkat lunak Anda dapat mengidentifikasi tingkat yang telah didaftarkan pelanggan dengan menjalankan operasi `CheckoutLicense` API dan menentukan semua tingkatan yang mungkin dalam permintaan.

Tanggapan permintaan berisi hak yang sesuai dengan tingkat yang telah diperoleh pelanggan. Berdasarkan informasi ini, perangkat lunak dapat memberikan pengalaman pelanggan yang sesuai.

Permintaan

```
linux-machine ~]$ aws license-manager checkout-license\
```

```
--product-sku "2205b290-19e6-4c76-9eea-377d6bf71a47" \  
--checkout-type "PROVISIONAL" \  
--key-fingerprint "aws:294406891311:AWS/Marketplace:issuer-fingerprint" \  
--entitlements "Name=BasicTier, Unit=None" "Name=IntermediateTier, Unit=None" \  
"Name=PremiumTier, Unit=None"
```

Respons

```
{  
  "CheckoutType": "PROVISIONAL",  
  "EntitlementsAllowed": [  
    {  
      "Name": "IntermediateTier",  
      "Units": "None"  
    }  
  ],  
  "Expiration": "2021-04-22T19:02:36",  
  "IssuedAt": "2021-04-22T18:02:36",  
  "LicenseArn": "arn:aws:license-manager::294406891311:license:l-16bf01b...",  
  "LicenseConsumptionToken": "AKIAIOSFODNN7EXAMPLE"  
}
```

Alur kerja integrasi

Langkah-langkah berikut menunjukkan alur kerja untuk mengintegrasikan produk AMI Anda dengan: AWS License Manager

1. Penjual menciptakan produk dengan AWS License Manager integrasi.
2. Penjual mencantumkan produk di AWS Marketplace.
3. Pembeli menemukan produk AWS Marketplace dan membelinya.
4. Lisensi dikirim ke pembeli di dalamnya Akun AWS.
5. Pembeli menggunakan perangkat lunak dengan meluncurkan instans Amazon Elastic Compute Cloud (Amazon EC2), Amazon Elastic Container Service (Amazon ECS) Service (Amazon ECS), atau tugas pod Amazon Elastic Kubernetes Service (Amazon EKS), Pelanggan menyebarkan dengan menggunakan peran IAM.
6. Perangkat lunak membaca lisensi di AWS License Manager akun pembeli, menemukan hak yang dibeli, dan menyediakan fitur yang sesuai.

Note

License Manager tidak melakukan pelacakan atau pembaruan apa pun; ini dilakukan oleh aplikasi penjual.

Prasyarat integrasi License Manager

Sebelum menerbitkan produk, Anda harus melakukan hal berikut:

1. Buat produk AMI baru di Portal Manajemen AWS Marketplace, dan catat kode produknya.
2. Isi Formulir Beban Produk (PLF) dengan informasi harga yang diperlukan, dan kembalikan kepada kami untuk diproses.
3. Gunakan peran IAM untuk tugas atau pod yang menjalankan aplikasi Anda dengan izin IAM yang diperlukan untuk memanggil `CheckoutLicense`, `ExtendLicenseConsumption` dan `CheckInLicense`

Izin IAM yang diperlukan dirinci dalam kebijakan IAM berikut.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "VisualEditor0",
      "Effect": "Allow",
      "Action": [
        "license-manager:CheckoutLicense",
        "license-manager:GetLicense",
        "license-manager:CheckInLicense",
        "license-manager:ExtendLicenseConsumption",
        "license-manager:ListReceivedLicenses"
      ],
      "Resource": "*"
    }
  ]
}
```

4. Lakukan panggilan uji ke operasi RegisterUsage API dengan catatan untuk semua dimensi harga yang Anda tentukan.

Mengintegrasikan produk berbasis AMI dengan AWS License Manager

Anda dapat mengintegrasikan produk berbasis AMI Anda dengan License Manager dengan menggunakan API. [AWS License Manager](#) Luncurkan EC2 instans Amazon dengan menggunakan produk berbasis AWS Marketplace AMI.

Note

Pastikan bahwa Anda telah menyelesaikan [the section called “Prasyarat integrasi License Manager”](#) sebelum Anda melakukan prosedur berikut.

Untuk mengintegrasikan produk berbasis AMI Anda dengan License Manager

1. Selesaikan prosedur dalam [the section called “Membuat lisensi pengujian di License Manager”](#). Anda harus membuat lisensi pengujian di License Manager untuk menguji integrasi Anda.
2. Jalankan operasi [GetLicense](#) API menggunakan lisensi Amazon Resource Name (ARN) yang Anda peroleh pada langkah 1. Perhatikan nilai KeyFingerprint atribut GetLicense respons untuk digunakan nanti.
3. Unduh dan sertakan AWS SDK publik terbaru dalam aplikasi Anda.
4. Untuk memverifikasi bahwa pembeli berhak menggunakan lisensi untuk aplikasi Anda, jalankan operasi [CheckoutLicense](#) API. Gunakan detail hak dan sidik jari kunci dari lisensi pengujian yang Anda peroleh di langkah 1.

Jika tidak ada hak yang ditemukan untuk lisensi, atau jumlah maksimum hak terlampaui, operasi API akan kembali. CheckoutLicense NoEntitlementsAllowedException Jika hak tersebut valid, atau tersedia untuk digunakan, CheckoutLicense operasi mengembalikan respons yang berhasil dengan hak yang diminta dan nilainya.

5. (Diperlukan hanya untuk hak mengambang) Jalankan operasi [CheckinLicense](#) API menggunakan LicenseConsumptionToken yang diterima dalam respons. CheckoutLicense Tindakan ini melepaskan hak yang telah diperiksa sebelumnya kembali ke kumpulan hak yang tersedia.
6. Setelah Anda berhasil memverifikasi integrasi License Manager dengan lisensi pengujian yang Anda buat pada langkah 1, perbarui sidik jari kunci dalam kode

Andaaws:294406891311:AWS/Marketplace:issuer-fingerprint. Sekarang, Anda siap untuk bekerja dengan lisensi yang dikeluarkan oleh AWS Marketplace.

Ikuti proses rilis pembuatan aplikasi ke produk AMI dan kemudian kirimkan produk untuk AWS Marketplace mengikuti proses penerbitan produk.

Membuat lisensi pengujian di License Manager

Anda menggunakan versi 2 dari AWS Command Line Interface (AWS CLI) untuk membuat lisensi pengujian di AWS License Manager. Lisensi pengujian ini hanya digunakan untuk memverifikasi dan menguji AWS License Manager integrasi. Setelah pengujian selesai, Anda dapat menghapus lisensi pengujian. Lisensi sebenarnya dihasilkan oleh AWS Marketplace dengan sidik jari kunci yang berbeda.

AWS Marketplace mendukung dua jenis hak di AWS License Manager Namun, hanya satu jenis yang dapat diaktifkan untuk suatu produk. Saat Anda membuat lisensi, termasuk lisensi pengujian, Anda harus menentukan salah satu jenis hak berikut:

Hak berjenjang - Model lisensi berjenjang memberikan hak kepada pelanggan untuk fitur aplikasi tertentu. Pelanggan tidak dapat menentukan jumlah unit yang ingin mereka beli. Namun, mereka dapat memilih satu paket atau tingkat yang telah ditentukan sebelumnya. Pelanggan dapat memodifikasi kontrak nanti untuk berlangganan ke tingkat lain.

Hak yang dapat dikonfigurasi — Model lisensi yang dapat dikonfigurasi memberikan hak untuk sejumlah sumber daya tertentu ketika pelanggan memperoleh lisensi. Pelanggan memilih jumlah unit yang ingin mereka beli selama proses berlangganan dan akan ditagih berdasarkan harga satuan. Pelanggan juga dapat berlangganan berbagai dimensi.

Parameter yang diperlukan untuk digunakan dalam operasi CheckoutLicense API adalah sebagai berikut:

- **CheckoutType**— Nilai yang valid adalah `Perpetual` atau `Provisional`:
 - **Perpetual**— Digunakan ketika jumlah hak yang diperiksa akan habis dari kolam renang. Contoh: Pembeli berhak memproses 500 GB data. Ketika mereka terus memproses data, kuantitas ditarik ke bawah dan habis dari kumpulan 500 GB. Mendapat status lisensi yang dibeli apakah lisensi tersebut kedaluwarsa atau akan kedaluwarsa untuk mengirim pemberitahuan kepada pelanggan.
 - **Provisional**— Digunakan untuk hak lisensi mengambang di mana hak diperiksa keluar dari kolam dan dikembalikan kembali setelah digunakan. Contoh: Pengguna berhak atas 500

pengguna simultan dalam aplikasi. Saat pengguna masuk dan keluar, pengguna ditarik dan dikembalikan ke kumpulan 500 pengguna. Untuk informasi selengkapnya tentang hak lisensi mengambang, lihat Lisensi yang [dikeluarkan Penjual di License Manager AWS](#).

- **ClientToken**— Pengidentifikasi unik dan peka huruf besar/kecil untuk memastikan hasil yang tepat terjadi dan sama tidak peduli berapa kali dicoba. Kami menyarankan Anda menggunakan pengenal unik universal acak (UUID) untuk setiap permintaan.
- **Entitlements**— Daftar hak yang akan diperiksa.
 - Untuk hak berjenjang, berikan Name dan Unit properti sebagai berikut:

```
{
  "Name": "<Entitlement_Name>",
  "Unit": "None"
}
```

- Untuk hak yang dapat dikonfigurasi, berikan, NameUnit, dan Value properti sebagai berikut:

```
{
  "Name": "<Entitlement_Name>",
  "Unit": "<Entitlement_Unit>",
  "Value": <Desired_Count>{
}
```

- **KeyFingerprint**— Gunakan sidik jari kunci ini untuk memverifikasi bahwa lisensi dikeluarkan oleh AWS Marketplace. Sidik jari kunci untuk lisensi yang dikeluarkan oleh AWS Marketplace adalah sebagai berikut:

```
aws:294406891311:AWS/Marketplace:issuer-fingerprint
```

- **Product SKU**— ID Produk dengan format Globally Unique Identifier (GUID) yang terkait dengan suatu AWS Marketplace produk.

Example dari hak yang dapat dikonfigurasi

Berikut ini adalah contoh permintaan yang menggunakan operasi CheckoutLicense API untuk memeriksa hak yang dapat dikonfigurasi bernama. PowerUsers

```
aws license-manager checkout-license \  
product-sku "2205b290-19e6-4c76-9eea-377d6bf71a47" \  
checkout-type "PROVISIONAL" \  
client-token "79464194dca9429698cc774587a603a1" \  
entitlements "Name=PowerUsers,Value=1,Unit=Count" \  
key-fingerprint "aws:294406891311:AWS/Marketplace:issuer-fingerprint"
```

Example dari hak berjenjang

Berikut ini adalah contoh permintaan yang menggunakan operasi CheckoutLicense API untuk memeriksa hak fitur bernama EnterpriseEdition.

```
aws license-manager checkout-license \  
--product-sku "2205b290-19e6-4c76-9eea-377d6bf71a47" \  
--checkout-type "PROVISIONAL" \  
--client-token "79464194dca9429698cc774587a603a1" \  
--entitlements "Name=EnterpriseEdition,Unit=None" \  
--key-fingerprint "aws:294406891311:AWS/Marketplace:issuer-fingerprint"
```

Untuk membuat lisensi uji untuk produk berbasis AMI Anda

1. Dari lingkungan lokal Anda dengan AWS CLI v2 diinstal, jalankan skrip berikut. Skrip membuat lisensi pengujian dan mengonfigurasi detail produk yang sesuai.

Note

Gunakan yang berbeda Akun AWS dari tes Akun AWS di mana Anda menyebarkan dan menguji perangkat lunak Anda. Lisensi tidak dapat dibuat, diberikan kepada, dan diperiksa dalam hal yang sama Akun AWS.

```
#!/bin/bash  
  
# Replace with intended product ID on AWS Marketplace  
PRODUCT_ID=<REPLACE-WITH-PRODUCT-ID>  
  
# Replace with license recipient's AWS Account ID  
BENEFICIARY_ACCOUNT_ID=<REPLACE-WITH-BENEFICIARY-ACCOUNT-ID>
```

```
# Replace with your product's name
PRODUCT_NAME="Test Product"

# Replace with your seller name on AWS Marketplace
SELLER_OF_RECORD="Test Seller"

# Replace with intended license name
LICENSE_NAME="AWSMP Test License"

# Replace the following with desired contract dimensions
# More info here: https://docs.aws.amazon.com/license-manager/latest/APIReference/API\_Entitlement.html
# Example "configurable entitlement"
ENTITLEMENTS='[
  {
    "Name": "ReadOnly",
    "MaxCount": 5,
    "Overage": false,
    "Unit": "Count",
    "AllowCheckIn": true
  }
]'
# Example "tiered entitlement"
# ENTITLEMENTS='[
#   {
#     "Name": "EnterpriseUsage",
#     "Value": "Enabled",
#     "Unit": "None"
#   }
# ]'

# Format "yyyy-mm-ddTHH:mm:ss.SSSZ"
# This creates a validity period of 10 days starting the current day
# Can be updated to desired dates
VALIDITY_START=$(date +%Y-%m-%dT%H:%M:%S.%SZ)
VALIDITY_END=$(date --date="+10 days" +%Y-%m-%dT%H:%M:%S.%SZ)

# Configuration for consumption of the license as set on Marketplace products
CONSUMPTION_CONFIG='{
  "RenewType": "None",
  "ProvisionalConfiguration": {
    "MaxTimeToLiveInMinutes": 60
  }
}'
```

```
# License's home Region
HOME_REGION=us-east-1

# License issuer's name
ISSUER=Self

# Run AWS CLI command to create a license
aws license-manager create-license \
  --license-name "${LICENSE_NAME}" \
  --product-name "${PRODUCT_NAME}" \
  --product-sku "${PRODUCT_ID}" \
  --issuer Name="${ISSUER}" \
  --beneficiary "${BENEFICIARY_ACCOUNT_ID}" \
  --validity 'Begin="""${VALIDITY_START}""',End="""${VALIDITY_END}""' \
  --entitlements "${ENTITLEMENTS}" \
  --home-region "${HOME_REGION}" \
  --region "${HOME_REGION}" \
  --consumption-configuration "${CONSUMPTION_CONFIG}" \
  --client-token $(uuidgen)
```

2. Berikan lisensi menggunakan AWS License Manager konsol. Untuk informasi selengkapnya, lihat [mendistribusikan hak](#) di Panduan Pengguna License Manager.
3. Masuk ke akun Akun AWS yang bertindak sebagai akun pembeli tempat Anda akan menyebarkan dan menguji perangkat lunak Anda. Ini harus berbeda Akun AWS dari Akun AWS yang dibuat dan diberikan lisensi.
4. Buka AWS License Manager konsol untuk menerima dan mengaktifkan lisensi yang diberikan. Untuk informasi selengkapnya, lihat [mengelola lisensi yang Anda berikan](#) di Panduan Pengguna License Manager.
5. Jalankan perintah berikut di lingkungan Anda.

```
# The following example uses a key fingerprint that should match the test license
you created.
# When checking out an actual AWS Marketplace created license, use the following
fingerprint:
# aws:294406891311:AWS/Marketplace:issuer-fingerprint
aws license-manager checkout-license \
  --product-sku <REPLACE-WITH-PRODUCT-ID> \
  --checkout-type PROVISIONAL \
  --key-fingerprint "aws:<ACCOUNT-ID-WHERE-YOU-CREATED-TEST-LICENSE>:Self:issuer-
fingerprint" \
```

```
--entitlements "Name=ReadOnly,Value=1,Unit=Count" \  
--client-token $(uuidgen)
```

Perintah sebelumnya digunakan PROVISIONAL sebagai nilai untuk CheckoutType parameter. Jika hak menggunakan lisensi drawdown, gunakan PERPETUAL untuk nilainya.

Panggilan API License Manager

Untuk mengelola lisensi yang disimpan di akun License Manager pelanggan, perangkat lunak Anda dapat menggunakan panggilan API berikut:

- **GetLicense**— Mendapat status lisensi yang dibeli apakah lisensi tersebut kedaluwarsa atau akan kedaluwarsa untuk mengirim pemberitahuan kepada pelanggan.
- **CheckoutLicense**— Menemukan lisensi yang telah dibeli pengguna. Anda juga dapat menggunakannya untuk memperbarui jumlah lisensi ketika pengguna telah menggunakan sejumlah lisensi. Dengan **CheckoutLicense**, Anda dapat terus memeriksa jumlah lisensi yang digunakan oleh pelanggan. Ketika pelanggan kehabisan semua lisensi, panggilan ini mengembalikan kesalahan. Untuk informasi tentang irama yang disarankan untuk dijalankan **CheckoutLicense**, lihat [the section called “Perpanjangan dan peningkatan lisensi”](#).
- **ExtendLicenseConsumption**— Dalam kasus dimensi mengambang, ketika perangkat lunak memeriksa lisensi, itu akan mengembalikan lisensi ke kolam secara otomatis setelah 60 menit. Jika Anda ingin memperpanjang waktu lisensi tetap diperiksa, perangkat lunak Anda dapat menelepon **ExtendLicenseConsumption** untuk memperpanjang lisensi selama 60 menit lagi.
- **CheckInLicense**— Dalam hal dimensi mengambang, ketika Anda ingin mengembalikan lisensi ke kumpulan hak, gunakan **CheckInLicense**.
- **ListReceivedLicenses**— Daftar lisensi yang dibeli oleh pembeli.

Perpanjangan dan peningkatan lisensi

Pelanggan dapat memperbarui atau meningkatkan lisensi mereka di Portal Manajemen AWS Marketplace Setelah mereka melakukan pembelian tambahan AWS Marketplace, buat versi baru dari lisensi yang mencerminkan hak baru. Perangkat lunak Anda membaca hak baru menggunakan panggilan API yang sama. Anda tidak perlu melakukan hal yang berbeda dalam hal Integrasi License Manager untuk menangani pembaruan dan peningkatan.

Karena perpanjangan lisensi, peningkatan, pembatalan, dan sebagainya, kami menyarankan agar produk Anda melakukan panggilan **CheckoutLicense** API dengan irama reguler saat produk

sedang digunakan. Dengan menggunakan operasi CheckoutLicense API pada irama reguler, produk dapat mendeteksi perubahan hak seperti upgrade dan kedaluwarsa.

Kami menyarankan Anda melakukan panggilan CheckoutLicense API setiap 15 menit.

Menerima notifikasi Amazon SNS untuk produk AMI AWS Marketplace

Untuk menerima pemberitahuan tentang perubahan langganan pelanggan untuk produk Anda, Anda dapat berlangganan topik Amazon Simple Notification Service (Amazon SNS) untuk AWS Marketplace diberikan kepada Anda selama pembuatan produk. Misalnya, Anda dapat mengetahui kapan pelanggan menerima penawaran pribadi. [Topik Amazon SNS: aws-mp-subscription-notification](#) adalah topik Amazon SNS yang tersedia untuk produk AMI. Topik ini memberi tahu Anda saat pembeli berlangganan atau berhenti berlangganan suatu produk. Pemberitahuan ini tersedia untuk model harga per jam, termasuk per jam dan per jam dengan tahunan. Untuk informasi selengkapnya, silakan lihat bagian-bagian berikut ini.

Note

Selama proses pembuatan produk, topik Amazon SNS dibuat untuk produk Anda. Untuk berlangganan notifikasi, Anda memerlukan Nama Sumber Daya Amazon (ARN) dari topik Amazon SNS (misalnya,). `arn:aws:sns:us-east-1:123456789012:aws-mp-subscription-notification-PRODUCTCODE` ARN tidak tersedia di portal penjual untuk produk server. Hubungi [tim AWS Marketplace operasi](#) untuk meminta ARN.

Topik

- [Topik Amazon SNS: aws-mp-subscription-notification](#)
- [Berlangganan antrian Amazon SQS ke topik Amazon SNS](#)

Topik Amazon SNS: aws-mp-subscription-notification

Setiap pesan dalam `aws-mp-subscription-notification` topik untuk `subscribe-fail` tindakan `subscribe-success` dan memiliki format berikut.

```
{
```

```
"action": "action-name",
"customer-identifier": " X01EXAMPLEX",
"product-code": "n0123EXAMPLEXXXXXXXXXXXXX",
"offer-identifier": "offer-abcexample123"
}
```

<*action-name*> Akan bervariasi tergantung pada notifikasi. Tindakan yang mungkin adalah:

- subscribe-success
- subscribe-fail
- unsubscribe-pending
- unsubscribe-success

`offer-identifier` Ini termasuk dalam pemberitahuan hanya ketika tindakan `subscribe-success` atau `subscribe-fail`. Itu tidak termasuk dalam notifikasi saat tindakan `unsubscribe-pending` atau `unsubscribe-success`. Untuk penawaran yang dibuat sebelum Januari 2024, pengenal ini disertakan dalam pemberitahuan hanya untuk penawaran pribadi. Untuk penawaran yang dibuat pada Januari 2024 dan yang lebih baru, pengenal ini disertakan dalam pemberitahuan untuk semua penawaran, termasuk penawaran pribadi dan penawaran publik.

Untuk informasi tentang jenis penawaran, lihat tanggapan dari [DescribeEntity API](#) atau visibilitas penawaran perjanjian di dasbor [perpanjangan Perjanjian](#).

Note

Untuk [DescribeEntity API](#), jika Anda menemukan aspek penargetan Akun AWS dalam akun dari aturan penargetan untuk penawaran itu, itu adalah penawaran pribadi. Jika tidak ada aspek penargetan Akun AWS dalam akun dari aturan penargetan untuk penawaran itu, itu adalah penawaran umum.

Berlangganan antrian Amazon SQS ke topik Amazon SNS

Sebaiknya berlangganan antrian Amazon SQS ke topik SNS yang disediakan. Untuk petunjuk mendetail tentang cara membuat antrian SQS dan berlangganan antrian ke topik, lihat [Berlangganan antrian Amazon SQS ke topik Amazon SNS di Panduan Pengembang Layanan Pemberitahuan Sederhana Amazon](#).

Note

Anda hanya dapat berlangganan topik AWS Marketplace SNS dari yang Akun AWS digunakan untuk menjual produk. Namun, Anda dapat meneruskan pesan ke akun lain. Untuk informasi selengkapnya, lihat [Mengirim pesan Amazon SNS ke antrean Amazon SQS di akun lain di Panduan Pengembang Layanan Pemberitahuan Sederhana Amazon](#).

Polling antrian SQS untuk pemberitahuan

Setelah Anda berlangganan antrian SQS Anda ke topik SNS, pesan disimpan dalam SQS. Anda harus menentukan layanan yang terus-menerus melakukan polling antrian, mencari pesan, dan menanganinya sesuai dengan itu.

Daftar periksa produk AMI untuk AWS Marketplace

Sebelum mengirimkan produk Amazon Machine Image (AMI) Anda AWS, gunakan daftar periksa ini untuk memvalidasi kiriman Anda. Mengikuti persyaratan ini akan membantu memastikan proses publikasi yang efisien.

Penggunaan produk

- AMI siap produksi.
- Penggunaan produk tidak dibatasi oleh waktu atau pengukuran lainnya.
- AMI kompatibel dengan pengalaman pemenuhan 1-klik.
- Segala sesuatu yang diperlukan untuk menggunakan produk ada dalam perangkat lunak, termasuk aplikasi klien. Produk yang memerlukan dependensi eksternal, seperti paket perangkat lunak atau aplikasi klien, mengikuti [Kebijakan penggunaan produk](#) yang menyertakan pengungkapan yang tepat.
- AMI memenuhi wajib [Persyaratan produk berbasis AMI untuk AWS Marketplace](#).
- Tidak ada lisensi tambahan yang diperlukan untuk menggunakan produk.
- Pembeli tidak harus memberikan informasi privat (misalnya, alamat email mereka) untuk menggunakan produk.

Persiapan AMI

- Nama dan deskripsi produk Anda harus sesuai dengan bidang Deskripsi produk AMI yang Anda berikan.
- Menggunakan virtualisasi Hardware Virtual Machine (HVM) dan arsitektur 64-bit.
- Produk tidak mengandung kerentanan, malware, atau virus yang diketahui.
- Pelanggan dapat mengakses instance melalui jaringan dan menggunakan akses administratif.
- Tidak ada rahasia hardcode yang ada di AMI. Contoh rahasia hardcode termasuk kata sandi default untuk pengguna dan layanan sistem, kunci pribadi, dan kredensial.
- Tidak ada kunci publik resmi SSH hardcode yang ada di AMI.
- Tes validasi Test 'Add Version' berhasil diselesaikan tanpa masalah.

Windows AMIs

- Untuk sistem operasi Windows Server 2012 dan yang lebih baru, AMI Anda menggunakan versi terbaru [EC2Launch v2](#).
- Jika Anda menggunakan EC2 Launch v2, lengkapi yang berikut ini:
 - Di [pengaturan EC2 Peluncuran Amazon](#), pilih Acak di bawah Setel akun administrator agar kata sandi administrator dibuat saat runtime.
 - Di [pengaturan Amazon EC2 Launch](#), pilih Aktifkan kembali dan mulai layanan SSM setelah Sysprep.
 - Tambahkan UserData ke [konfigurasi tugas EC2 v2](#).
- Untuk Windows Server 2012 dan versi yang lebih baru, hindari menggunakan [EC2Config](#). Jika EC2 Config diperlukan, pastikan Anda menggunakan versi terbaru.
- Jika Anda menggunakan EC2 Config, aktifkan parameter berikut dalam [file pengaturan](#) untuk AMI Anda:
 - Ec2SetPassword
 - Ec2WindowsActivate
 - Ec2HandleUserData
- Pastikan tidak ada akun tamu atau pengguna desktop jarak jauh.

Linux AMIs

- Login jarak jauh sebagai superuser dilarang.
- Akses jarak jauh berbasis kata sandi dilarang.

Formulir beban produk atau tab Produk

- Semua bidang yang diperlukan selesai.
- Semua nilai berada dalam batas karakter yang ditentukan.
- Semua URLs beban tanpa kesalahan.
- Gambar produk memiliki lebar setidaknya 110 piksel dan antara rasio 1:1 dan 2:1.
- Harga ditentukan untuk semua jenis instans yang diaktifkan (untuk harga bulanan per jam, berdasarkan per jam, dan model harga tahunan berbasis per jam).
- Harga bulanan ditentukan (untuk model harga bulanan dan bulanan berbasis per jam).

Persyaratan produk berbasis AMI untuk AWS Marketplace

AWS Marketplace mempertahankan kebijakan berikut untuk semua produk dan penawaran Amazon Machine Image (AMI). Kebijakan di bagian ini dimaksudkan untuk menyediakan platform komputasi yang aman, terjamin, dan dapat dipercaya kepada pelanggan.

Semua produk dan metadata terkait ditinjau ketika dikirimkan untuk memastikan mereka memenuhi atau melampaui kebijakan saat ini AWS Marketplace . Kebijakan ini diperbarui secara berkala agar selaras dengan pedoman keamanan yang berkembang. AWS Marketplace terus memindai produk untuk memverifikasi bahwa daftar yang ada terus memenuhi setiap perubahan pada persyaratan ini. Jika suatu produk tidak sesuai, AWS Marketplace akan menghubungi penjual untuk memperbarui produk mereka untuk memenuhi standar baru. Dalam beberapa kasus, produk mungkin dibuat sementara tidak tersedia untuk pelanggan baru sampai masalah diselesaikan. Proses ini membantu menjaga keamanan dan kepercayaan AWS Marketplace platform untuk semua pengguna.

Sebelum mengirimkan produk Anda, kami sangat menyarankan untuk menggunakan fitur [Uji 'Tambahkan Versi'](#) Portal Manajemen AWS Marketplace untuk memastikan kepatuhan terhadap kebijakan saat ini.

Topik

- [Kebijakan penjual produk AMI](#)

- [Kebijakan Keamanan](#)
- [Kebijakan arsitektur](#)
- [Petunjuk penggunaan produk AMI](#)
- [Kebijakan versi produk AMI](#)
- [Kebijakan informasi pelanggan](#)
- [Kebijakan penggunaan produk](#)

Kebijakan penjual produk AMI

Semua AMIs harus mematuhi kebijakan penjual berikut:

- Secara default, AWS Marketplace penjual dibatasi hingga maksimal 75 daftar produk AMI publik. Semua penjual di atas batas mereka tunduk pada tinjauan kinerja berkala dan mungkin diminta untuk membatasi daftar yang berkinerja buruk. AWS Marketplace dapat memberikan dan mencabut kenaikan batas ini atas kebijakannya sendiri.

Kebijakan Keamanan

Kebijakan umum

Semua AMIs harus mematuhi kebijakan berikut:

- AMIs harus lulus semua pemeriksaan keamanan yang dilakukan oleh [alat pemindaian AWS Marketplace AMI](#), tidak menunjukkan kerentanan atau malware yang diketahui.
- AMIs harus menggunakan sistem operasi dan perangkat lunak yang didukung saat ini. Sistem operasi dan perangkat lunak yang mencapai akhir masa pakainya tidak diperbolehkan.
- Otentikasi berbasis kata sandi untuk layanan contoh dilarang. Ini berlaku bahkan jika kata sandi dibuat, diatur ulang, atau ditentukan oleh pengguna saat peluncuran. Kata sandi nol dan kosong tidak diperbolehkan.

Pengecualian:

- Kata sandi administrator yang dihasilkan oleh EC2Config/EC2Launch pada instance Windows.
- Akses non-administratif ke layanan host (misalnya, aplikasi web) tanpa adanya metode otentikasi lainnya. Jika kata sandi yang kuat digunakan, mereka harus dibuat secara acak untuk setiap instance, digunakan sekali oleh administrator layanan untuk otentikasi awal, dan segera diubah setelah login pertama.

- AMI tidak boleh mengandung rahasia hardcode seperti pengguna sistem dan kata sandi layanan (termasuk kata sandi yang di-hash), kunci pribadi, atau kredensial.
- AMIs tidak boleh meminta AWS kredensial untuk mengakses AWS layanan. Jika produk Anda memerlukan akses ke AWS layanan, instance harus diberi peran dengan hak istimewa minimal AWS Identity and Access Management (IAM). Pengguna dapat membuat peran secara manual atau dengan menggunakan AWS CloudFormation template. Ketika peluncuran Single-AMI diaktifkan untuk produk dengan metode CloudFormation pengiriman, petunjuk penggunaan harus menyertakan panduan yang jelas untuk membuat peran IAM dengan hak istimewa minimal. Untuk informasi selengkapnya, lihat [Mengirimkan produk berbasis AMI Anda](#) menggunakan AWS CloudFormation
- Penjual tidak boleh memiliki akses ke instans yang dijalankan oleh pelanggan. Dalam hal akses tersebut diperlukan untuk dukungan atau tujuan lain, pelanggan dapat diinstruksikan untuk secara eksplisit mengaktifkannya.

Kebijakan akses SSH (Secure Shell)

Selain [kebijakan umum](#), AMIs menyediakan akses SSH (Secure Shell) harus mematuhi kebijakan keamanan berikut:

- AMIs tidak boleh mengizinkan otentikasi berbasis kata sandi menggunakan SSH. Untuk memastikan ini, dalam `sshd_config` file Anda, atur `PasswordAuthentication` ke `no`.
- AMIs harus menonaktifkan login jarak jauh berbasis kata sandi untuk akun pengguna super. Untuk informasi selengkapnya, lihat [Nonaktifkan login jarak jauh berbasis kata sandi untuk](#) pengguna `root`.
- AMIs tidak boleh berisi kunci publik resmi untuk akses SSH.
- SSH on AMIs harus dapat diakses oleh prosedur pemeriksaan AWS Marketplace internal.
 - Layanan SSH harus mendengarkan pada port TCP yang ditentukan untuk pemindaian AMI. Untuk informasi selengkapnya, lihat [Tambahkan versi baru](#).
 - SSH harus dapat diakses dari subnet `10.0.0.0/16` dan `10.2.0.0/16` pada alamat IP yang ditetapkan oleh Amazon Elastic Compute Cloud (Amazon EC2) saat peluncuran instans.

Kebijakan untuk AMIs berbasis Linux dan sistem operasi mirip Unix lainnya

Selain [kebijakan umum](#), AMIs berdasarkan Linux dan sistem operasi mirip Unix lainnya harus mematuhi kebijakan keamanan berikut:

- AMIs harus memungkinkan pengguna untuk mendapatkan akses yang sepenuhnya istimewa (misalnya, untuk mengizinkan sudo akses).

Kebijakan untuk berbasis Windows AMIs

Selain [kebijakan umum](#), berbasis Windows AMIs harus mematuhi kebijakan keamanan berikut:

- AMIs tidak boleh berisi akun tamu.
- Hanya akun administrator yang dapat diberikan akses desktop jarak jauh ke sebuah instans.
- Windows AMIs harus menghasilkan kata sandi administrator dengan mengaktifkan opsi ini di [EC2Launch](#) (atau [EC2Config](#) untuk Windows 2016 dan yang lebih lama):
 - `Ec2SetPassword`
 - `Ec2WindowsActivate`
 - `Ec2HandleUserData`
- AMIs harus tersedia untuk pemeriksaan otomatis. Setidaknya satu dari persyaratan berikut harus diterapkan:
 - (Opsi yang disarankan) Agen SSM diinstal dan memiliki izin administratif dan akses jaringan keluar.
 - [Layanan Windows Remote Management \(WinRM\)](#) diaktifkan, mendengarkan pada port TCP5985, dan dapat diakses dari subnet `10.0.0.0/16` dan `10.2.0.0/16` pada alamat IP yang ditetapkan oleh Amazon Elastic Compute Cloud (Amazon) saat peluncuran instans. EC2
 - [Layanan Protokol Microsoft Server Message Block \(SMB\) dan Common Internet File System \(CIFS\)](#) diaktifkan, mendengarkan pada port TCP dan dapat diakses dari subnet `139.44.5.10.0.0.0/16` dan `10.2.0.0/16` pada alamat IP yang ditetapkan oleh Amazon Elastic Compute Cloud (Amazon) saat peluncuran instans. EC2

Kebijakan arsitektur

Semua AMIs harus mematuhi kebijakan arsitektur berikut:

- Sumber AMIs untuk AWS Marketplace harus disediakan di Wilayah AS Timur (Virginia N.).
- AMIs harus menggunakan virtualisasi HVM.
- AMIs harus menggunakan arsitektur ARM x86-64 atau 64-bit.

- AMIs harus didukung oleh Amazon Elastic Block Store (Amazon EBS). Kami tidak mendukung AMIs didukung oleh Amazon Simple Storage Service.
- AMIs tidak boleh menggunakan snapshots EBS terenkripsi.
- AMIs tidak boleh menggunakan sistem file terenkripsi.
- AMIs harus dibangun sehingga mereka dapat berjalan di semua Wilayah AWS dan bersifat Region-agnostik. AMIs dibangun secara berbeda untuk Wilayah yang berbeda tidak diizinkan.

Petunjuk penggunaan produk AMI

Saat membuat petunjuk penggunaan untuk produk AMI Anda, ikuti langkah-langkah dan panduan yang ada di [the section called “Petunjuk penggunaan produk AMI dan kontainer”](#).

Kebijakan versi produk AMI

AWS Marketplace mengotomatiskan pengalaman manajemen versi untuk AWS pelanggan dan penjual menggunakan S-AMI, AMI dengan CloudFormation template, dan produk kontainer. Dengan arsip versi otomatis, versi produk apa pun yang telah dibatasi oleh penjual selama lebih dari dua tahun akan diarsipkan secara otomatis. Versi yang diarsipkan tidak lagi tersedia untuk diluncurkan AWS Marketplace untuk pelanggan baru, namun pengguna yang ada dapat terus menggunakan versi yang diarsipkan melalui templat peluncuran dan grup Amazon Auto EC2 Scaling dengan menentukan ID AMI. Versi arsip apa pun yang belum digunakan untuk meluncurkan instance baru dalam 13 bulan terakhir akan dihapus. Setelah versi yang diarsipkan dihapus, itu tidak lagi tersedia untuk diluncurkan untuk pengguna baru atau yang sudah ada.

Kebijakan informasi pelanggan

Semua AMIs harus mematuhi kebijakan informasi pelanggan berikut:

- Perangkat lunak tidak boleh mengumpulkan atau mengekspor data pelanggan tanpa sepengetahuan pelanggan dan persetujuan tegas kecuali sebagaimana disyaratkan oleh BYOL (Bring Your Own License). Aplikasi yang mengumpulkan atau mengekspor data pelanggan harus mengikuti pedoman ini:
 - Pengumpulan data pelanggan harus swalayan, otomatis, dan aman. Pembeli tidak perlu menunggu penjual menyetujui untuk menyebarkan perangkat lunak.
 - Pengumpulan data pelanggan harus konsisten dengan perjanjian Anda dengan AWS, termasuk namun tidak terbatas pada, [Syarat dan Ketentuan AWS Marketplace](#), [Ketentuan AWS Layanan](#), [Pemberitahuan AWS Privasi](#), dan [Perjanjian AWS Pelanggan](#).

- Informasi pembayaran tidak boleh dikumpulkan.

Kebijakan penggunaan produk

Semua AMIs harus mematuhi kebijakan penggunaan produk berikut:

- Produk tidak boleh membatasi akses ke fungsionalitas produk atau produk berdasarkan waktu, jumlah pengguna, atau pembatasan lainnya. Produk beta dan prarilis, atau produk yang tujuan utamanya adalah menawarkan fungsionalitas uji coba atau evaluasi, tidak didukung. Pengembang, Komunitas, dan BYOL edisi perangkat lunak komersial didukung, asalkan versi berbayar setara juga tersedia di AWS Marketplace.
- Semua AMIs harus kompatibel dengan pengalaman Peluncuran dari Situs Web atau pengiriman berbasis AMI melalui AWS CloudFormation Untuk Peluncuran dari Situs Web, AMI tidak dapat meminta data pelanggan atau pengguna pada pembuatan instance untuk berfungsi dengan benar.
- AMIs dan perangkat lunak mereka harus dapat digunakan dengan cara swalayan dan tidak boleh memerlukan metode atau biaya pembayaran tambahan. Aplikasi yang memerlukan dependensi eksternal pada penerapan harus mengikuti pedoman ini:
 - Persyaratan harus diungkapkan dalam deskripsi atau petunjuk penggunaan daftar. Misalnya, Produk ini memerlukan koneksi internet untuk digunakan dengan benar. Paket-paket berikut diunduh saat penerapan: <list of package>
 - Penjual bertanggung jawab atas penggunaan dan memastikan ketersediaan dan keamanan semua dependensi eksternal.
 - Jika dependensi eksternal tidak lagi tersedia, produk harus dihapus AWS Marketplace juga.
 - Dependensi eksternal tidak boleh memerlukan metode atau biaya pembayaran tambahan.
- AMIs yang memerlukan koneksi berkelanjutan ke sumber daya eksternal yang tidak berada di bawah kendali langsung pembeli — misalnya, eksternal APIs atau Layanan AWS dikelola oleh penjual atau pihak ketiga — harus mengikuti pedoman ini:
 - Persyaratan harus diungkapkan dalam deskripsi atau petunjuk penggunaan daftar. Misalnya, Produk ini membutuhkan koneksi internet yang berkelanjutan. Layanan eksternal yang sedang berlangsung berikut ini diperlukan untuk berfungsi dengan baik: <list of resources>
 - Penjual bertanggung jawab atas penggunaan dan memastikan ketersediaan dan keamanan semua sumber daya eksternal.
 - Jika sumber daya eksternal tidak lagi tersedia, produk harus dihapus AWS Marketplace juga.

- Sumber daya eksternal tidak boleh memerlukan metode atau biaya pembayaran tambahan dan pengaturan koneksi harus otomatis.
- Perangkat lunak produk dan metadata tidak boleh berisi bahasa yang mengarahkan pengguna ke platform cloud lain, produk tambahan, atau layanan upsell yang tidak tersedia di AWS Marketplace.
- Jika produk Anda merupakan add-on untuk produk lain atau produk ISV lain, deskripsi produk Anda harus menunjukkan bahwa itu memperluas fungsionalitas produk lain dan bahwa tanpa itu, produk Anda memiliki utilitas yang sangat terbatas. Misalnya, Produk ini memperluas fungsionalitas dan tanpa itu, produk ini memiliki utilitas yang sangat terbatas <product name>. Harap dicatat bahwa mungkin memerlukan lisensi sendiri untuk fungsionalitas penuh dengan daftar ini. <product name>

EC2 Produk berbasis komponen Image Builder di AWS Marketplace

Sebagai AWS Marketplace penjual, Anda dapat mengirimkan produk Anda ke pembeli dengan [komponen Image Builder](#). Komponen Image Builder adalah blok bangunan dasar yang digunakan dalam layanan Image Builder untuk membuat dan memelihara komponen EC2 Image Builder yang disesuaikan. Komponen dapat digunakan kembali, potongan konfigurasi modular yang menentukan serangkaian tugas tertentu yang akan dilakukan selama proses pembuatan gambar.

Komponen mendefinisikan urutan langkah yang diperlukan untuk menyesuaikan instance sebelum pembuatan gambar (komponen build), atau untuk menguji instance yang diluncurkan dari gambar yang dibuat (komponen pengujian). Komponen pembuat gambar dapat terdiri dari tiga item:

- Dokumen komponen, file deklaratif, YAMAL, atau JSON yang menentukan langkah dan tindakan yang akan dijalankan saat komponen digunakan untuk membuat gambar.
- Atribut komponen yang menyediakan opsi konfigurasi untuk komponen. Mereka membantu menentukan karakteristik, perilaku, dan kompatibilitas komponen.
- Paket perangkat lunak, dependensi yang harus diinstal agar komponen berfungsi dengan benar.

Penerbitan komponen Image Builder aktif AWS Marketplace didukung menggunakan AWS Marketplace Catalog API. Untuk petunjuk memublikasikan komponen Anda menggunakan Catalog API, lihat [produk komponen Bekerja dengan EC2 Image Builder](#) di Referensi AWS Marketplace API.

Membuat permintaan perubahan untuk produk EC2 Image Builder di AWS Marketplace

Untuk membuat perubahan pada produk atau versi AWS Marketplace, Anda mengirimkan permintaan perubahan melalui Portal Manajemen AWS Marketplace. Permintaan perubahan ditambahkan ke antrean dan dapat memakan waktu dari hitungan menit hingga hitungan untuk diselesaikan, tergantung tipe permintaannya. Anda dapat melihat status permintaan di Portal Manajemen AWS Marketplace. Topik ini menyediakan prosedur yang dapat Anda gunakan untuk membuat permintaan perubahan untuk satu produk AWS Marketplace, termasuk opsi untuk menggunakan pengalaman swalayan.

Anda dapat membuat permintaan perubahan untuk situasi berikut:

- Anda menyimpan langkah-langkah yang sedang berlangsung, tetapi tidak menyelesaikan seluruh proses, saat menggunakan pengalaman layanan mandiri untuk membuat daftar produk tunggal. Untuk menyelesaikan langkah-langkah yang tersisa, Anda membuat permintaan perubahan.
- Anda ingin melakukan modifikasi pada informasi produk untuk produk Anda yang berada dalam keadaan Terbatas atau Publik. Untuk memperbarui informasi, Anda membuat permintaan perubahan. Untuk informasi selengkapnya tentang jenis perubahan yang dapat Anda minta untuk produk EC2 Image Builder, lihat [Buat permintaan perubahan](#).

Note

Selain itu Portal Manajemen AWS Marketplace, Anda juga dapat membuat permintaan perubahan dengan menggunakan [API AWS Marketplace Katalog](#).

Topik

- [Buat permintaan perubahan dengan menggunakan layanan mandiri](#)
- [Buat permintaan perubahan](#)
- [Dapatkan status permintaan perubahan](#)
- [Sumber daya tambahan](#)

Buat permintaan perubahan dengan menggunakan layanan mandiri

Untuk membuat modifikasi pada versi atau informasi produk, Anda membuat permintaan perubahan di Portal Manajemen AWS Marketplace. Permintaan perubahan adalah blok bangunan dari daftar layanan mandiri yang Anda gunakan untuk membuat perubahan pada produk Anda. Setiap kali Anda memilih Simpan dan keluar dari langkah-langkah atau pilih Kirim untuk pembaruan apa pun, Anda membuat permintaan perubahan. Anda dapat menemukan permintaan Anda di tab Portal Manajemen AWS Marketplace [Permintaan](#).

Untuk membuat permintaan perubahan menggunakan swalayan

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/>, dan masuk ke akun penjual Anda, lalu buka halaman [Produk Server](#).
2. Pada tab Produk Server, pilih produk yang ingin Anda modifikasi.
3. Pilih satu opsi dari daftar tarik turun Permintaan perubahan.

4. Setelah Anda membuat permintaan perubahan, ada waktu tunggu untuk sistem memproses permintaan Anda, tercermin Dalam Tinjauan. Ketika permintaan selesai, itu akan menghasilkan Berhasil atau Gagal.
 - Setelah permintaan diajukan, permintaan mulai diproses melalui status ini: Sedang ditinjau, Mempersiapkan perubahan, dan Menerapkan perubahan.
 - Berhasil berarti bahwa perubahan yang diminta telah diproses dan perubahan tercermin dalam sistem.
 - Gagal berarti ada yang tidak beres dengan permintaan, sehingga perubahan tidak diproses. Jika status Gagal, Anda dapat memilih permintaan untuk menemukan kode kesalahan yang memberikan rekomendasi tentang cara memperbaiki kesalahan. Pada titik ini, Anda dapat memecahkan masalah kesalahan dan membuat permintaan baru untuk perubahan. Untuk membuat proses lebih cepat, Anda dapat memilih Salin ke permintaan baru untuk menyalin rincian permintaan yang gagal. Kemudian, Anda dapat melakukan penyesuaian dan mengirimkan kembali permintaan.

Buat permintaan perubahan

Important

Pada 15 Juni 2023, AWS Marketplace akan menghentikan prosedur berikut. Setelah 15 Juni 2023, gunakan [the section called “Buat permintaan perubahan dengan menggunakan layanan mandiri”](#) prosedur ini.

Untuk membuat modifikasi pada versi atau informasi produk, Anda membuat permintaan perubahan di Portal Manajemen AWS Marketplace.

Untuk membuat permintaan perubahan

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/>, dan masuk ke akun penjual Anda, lalu buka halaman [Produk Server](#).
2. Pada tab Produk Server, pilih produk yang ingin Anda modifikasi.
3. Pilih opsi dari daftar tarik-turun Permintaan perubahan.

Untuk sebagian besar permintaan perubahan, isi formulir di antarmuka pengguna dan kirimkan.

Untuk informasi selengkapnya tentang status permintaan perubahan, lihat [the section called “Dapatkan status permintaan perubahan”](#). Untuk mengetahui potensi masalah dengan permintaan perubahan, lihat [Memecahkan masalah kesalahan umum untuk permintaan perubahan pada AWS Marketplace](#).

Dapatkan status permintaan perubahan

Important

Pada 15 Juni 2023, AWS Marketplace akan menghentikan prosedur berikut. Prosedur ini tidak lagi diperlukan untuk pengalaman swalayan.

Setelah Anda mengirimkan permintaan perubahan, Anda dapat melihat status permintaan Anda dari tab **Permintaan** pada halaman [Produk Server](#) Portal Manajemen AWS Marketplace. Status dapat merupakan hal berikut:

- Dalam tinjauan berarti bahwa permintaan Anda sedang ditinjau. Beberapa permintaan memerlukan peninjauan manual oleh AWS Marketplace tim tetapi sebagian besar ditinjau secara otomatis dalam sistem.
- Berhasil berarti bahwa permintaan Anda sudah lengkap. Produk atau versi Anda telah diperbarui sesuai permintaan.
- Butuh tindakan berarti Anda perlu memperbarui permintaan Anda untuk memperbaiki masalah atau menjawab pertanyaan tentang permintaan tersebut. Pilih permintaan untuk melihat detailnya, termasuk masalah apa pun.
- Gagal berarti ada yang tidak beres dengan permintaan tersebut, dan Anda harus membuat permintaan baru untuk perubahan tersebut, dengan data yang sama.

Sumber daya tambahan

Untuk detail selengkapnya tentang permintaan perubahan untuk jenis pembaruan tertentu, lihat sumber daya berikut:

- [Memperbarui informasi produk EC2 Image Builder AWS Marketplace](#)

Memperbarui visibilitas produk EC2 Image Builder

Sebagai AWS Marketplace penjual, Anda dapat memperbarui visibilitas produk Anda untuk mengubah pembeli mana yang dapat melihat produk Anda. AWS Marketplace Jika status visibilitas disetel ke Publik, produk Anda dapat dilihat oleh semua AWS Marketplace pembeli. Jika visibilitas produk Anda diatur ke Limited, produk Anda hanya dapat dilihat oleh daftar Akun AWS IDs yang Anda izinkan. Anda juga dapat mengelola dan memperbarui daftar yang diizinkan ini Akun AWS IDs yang dapat melihat produk Anda. Bagian berikut menunjukkan cara memperbarui visibilitas produk dan daftar izin visibilitas terbatas.

Topik

- [Perbarui visibilitas produk](#)
- [Perbarui daftar yang diizinkan \(akun pratinjau\)](#)

Perbarui visibilitas produk

Untuk mengubah pembeli mana yang dapat melihat produk Anda AWS Marketplace, Anda dapat menggunakan Perbarui visibilitas.

Untuk memperbarui visibilitas

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/>, lalu masuk ke akun penjual Anda.
2. Buka halaman [Produk Server](#), pada tab Produk server saat ini, pilih produk yang ingin Anda modifikasi.
3. Dari menu tarik-turun Permintaan perubahan, pilih Perbarui visibilitas.

Note

Anda dapat meminta agar produk dipindahkan dari status Terbatas ke status Publik dengan menggunakan permintaan perubahan ini. Namun, permintaan perubahan harus melalui proses persetujuan tim Operasi AWS Marketplace Penjual untuk dipindahkan ke Publik.

4. Pilih Kirim permintaan perubahan untuk mengirimkan permintaan Anda untuk ditinjau.
5. Verifikasi bahwa tab Permintaan menampilkan status Permintaan sebagai Dalam peninjauan. Ketika permintaan selesai, status menjadi Berhasil.

Perbarui daftar yang diizinkan (akun pratinjau)

Untuk mengubah daftar Akun AWS IDs yang dapat melihat produk Anda dalam keadaan Terbatas, gunakan Perbarui daftar yang diizinkan.

Untuk memperbarui daftar yang diizinkan

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/>, lalu masuk ke akun penjual Anda.
2. Buka halaman [Produk Server](#), dan pada tab Produk server saat ini, pilih produk yang ingin Anda modifikasi.
3. Dari menu tarik-turun Permintaan perubahan, pilih Perbarui daftar yang diizinkan. Daftar saat ini disediakan dengan daftar akun yang saat ini diizinkan terdaftar.
4. Tambahkan Akun AWS IDs yang lebih disukai untuk visibilitas dan pisahkan IDs dengan koma.
5. Pilih Kirim permintaan perubahan untuk mengirimkan permintaan Anda untuk ditinjau.
6. Verifikasi bahwa tab Permintaan menampilkan status Permintaan sebagai Dalam peninjauan. Ketika permintaan selesai, status menjadi Berhasil.

Menambahkan dan membatasi EC2 instans Amazon untuk AWS Marketplace

Sebagai AWS Marketplace penjual, Anda dapat mengelola contoh mana yang dapat digunakan pembeli. Anda dapat menambahkan contoh baru untuk produk Anda yang dapat digunakan pembeli. Demikian pula, jika Anda ingin mencegah pembeli baru menggunakan produk Anda dari contoh tertentu, Anda dapat membatasi instance tersebut.

Untuk informasi selengkapnya tentang jenis EC2 instans Amazon, lihat [Jenis instans yang tersedia](#) di Panduan EC2 Pengguna Amazon.

Bagian berikut menjelaskan cara menambahkan dan membatasi instance.

Topik

- [Menambahkan sebuah instance](#)
- [Membatasi sebuah instance](#)

Menambahkan sebuah instance

Anda dapat menambahkan instance baru yang dapat digunakan pembeli sebagai komponen EC2 Image Builder.

Untuk menambahkan sebuah instance

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/>, lalu masuk ke akun penjual Anda.
2. Buka halaman [Produk Server](#), dan pada tab Produk server saat ini, pilih produk yang ingin Anda modifikasi.
3. Dari menu tarik-turun Permintaan perubahan, pilih Tambahkan instance.
4. Pilih arsitektur instance.
5. Pilih jenis instance yang ingin Anda tambahkan dari daftar instance yang tersedia.
6. Pilih Kirim permintaan untuk mengirimkan permintaan Anda untuk ditinjau.
7. Verifikasi bahwa tab Permintaan menampilkan status Permintaan sebagai Dalam peninjauan. Ketika permintaan selesai, status menjadi Berhasil.

Note

- Jika model harga Anda saat ini tidak gratis atau menggunakan model Bring Your Own License (BYOL), Anda juga harus menambahkan harga.
- Jika Anda membuat instans Tambah dengan harga untuk instans baru atau Perbarui harga untuk menaikkan harga, Anda tidak dapat menggunakan layanan mandiri untuk Menambahkan instans dalam 90 hari mulai dari hari Anda membuat perubahan. Untuk melakukan perubahan ini, hubungi [tim Operasi AWS Marketplace Penjual](#).
- Saat Anda menambahkan dukungan untuk jenis instans baru, pelanggan yang sudah berlangganan penawaran pribadi untuk produk Anda tidak akan dapat meluncurkan instance yang baru ditambahkan secara otomatis. Anda harus membuat penawaran pribadi lain dengan contoh yang Anda ingin pelanggan akses. Setelah menerima penawaran baru, pelanggan dapat meluncurkan instance yang baru ditambahkan. Pelanggan yang berlangganan produk Anda di masa mendatang juga dapat meluncurkan instance, selama instance tersebut termasuk dalam penawaran pribadi. Untuk informasi selengkapnya tentang cara membuat penawaran pribadi baru, lihat [Mengubah perjanjian di AWS Marketplace](#) nanti di panduan ini.

Membatasi sebuah instance

Untuk mencegah pembeli baru menggunakan instance, Anda dapat membatasi instans. Anda dapat menambahkan instance kembali di lain waktu, jika diperlukan. Pengguna komponen EC2 Image Builder yang ada pada instans terbatas dapat terus menggunakan produk dari Wilayah selama langganan mereka.

Untuk membatasi sebuah instance

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/>, lalu masuk ke akun penjual Anda.
2. Buka halaman [Produk Server](#), dan pada tab Produk server saat ini, pilih produk yang ingin Anda modifikasi.
3. Dari menu tarik-turun Permintaan perubahan, pilih Batasi instance.
4. Pilih contoh yang ingin Anda batasi, dan pilih Batasi.
5. Pilih Kirim permintaan perubahan untuk mengirimkan permintaan Anda untuk ditinjau.
6. Verifikasi bahwa tab Permintaan menampilkan status Permintaan sebagai Dalam peninjauan. Ketika permintaan selesai, status menjadi Berhasil.

Note

Jika kotak centang diarsir, ini berarti instance dikaitkan dengan satu hingga beberapa versi sebagai jenis instance yang direkomendasikan. Untuk membatasi instance tersebut, gunakan versi Perbarui untuk memilih jenis instans yang direkomendasikan berbeda. Setelah permintaan perubahan selesai dan instance yang ingin Anda batasi tidak lagi menjadi tipe instans yang direkomendasikan, Anda dapat kembali ke Batasi instance untuk membatasi instance yang Anda pilih.

Mengelola versi untuk produk EC2 Image Builder di AWS Marketplace

Manajemen versi untuk komponen EC2 Image Builder tidak dapat dilakukan dari UI. Untuk informasi selengkapnya, lihat bagian API katalog berikut:

- [Tambahkan versi baru](#)

- [Perbarui informasi tentang versi yang ada](#)
- [Batasi versi](#)

Memperbarui informasi produk EC2 Image Builder AWS Marketplace

Setelah membuat komponen EC2 Image Builder tunggal, Anda dapat mengubah beberapa informasi yang terkait dengannya AWS Marketplace. Misalnya, jika versi baru memodifikasi deskripsi atau sorotan produk, Anda dapat mengedit informasi produk dengan data baru. Anda juga dapat memperbarui informasi produk lainnya, termasuk judul produk, deskripsi SKU, kategori, kata kunci, dan lainnya. Untuk informasi lebih lanjut, lihat prosedur berikut.

Untuk memperbarui informasi produk

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/>, lalu masuk ke akun penjual Anda.
2. Buka halaman [Produk Server](#), dan pada tab Produk Server, pilih produk yang ingin Anda modifikasi.
3. Dari menu menurun Minta perubahan, pilih Perbarui informasi produk.
4. Perbarui salah satu bidang berikut yang perlu Anda ubah:
 - Judul produk
 - SKU
 - Deskripsi singkat
 - Deskripsi panjang
 - URL gambar logo produk
 - Sorotan
 - Kategori produk
 - Kata kunci
 - URL video produk
 - Sumber Daya
 - Informasi Support

 Note

Untuk detail tentang format logo, lihat [Persyaratan logo perusahaan dan produk](#).

5. Pilih Kirim.
6. Verifikasi bahwa permintaan muncul di tab Permintaan dengan status Dalam tinjauan. Anda mungkin perlu me-refresh halaman untuk melihat permintaan pada daftar.

Mengelola ketersediaan produk EC2 Image Builder menurut Wilayah AWS dan negara

Saat Anda membuat produk AWS Marketplace, Anda memilih AWS Wilayah di mana produk tersebut tersedia. Anda juga memilih negara tempat pembeli dapat membeli produk Anda. Kedua properti ini serupa, tetapi tidak sama. Misalnya, pembeli mungkin berlokasi di, dan membeli dari, Amerika Serikat, tetapi mereka mungkin berencana untuk menginstal produk Anda di Wilayah Eropa (Frankfurt). Agar pembeli ini dapat membeli produk Anda, Anda harus memasukkan Amerika Serikat dalam daftar negara Anda, dan Wilayah Eropa (Frankfurt) dalam daftar Wilayah Anda. Anda dapat menggunakan bagian berikut untuk memperbarui ketersediaan produk menurut Wilayah dan negara.

Topik

- [Tambahkan sebuah Wilayah AWS](#)
- [Membatasi sebuah Wilayah AWS](#)
- [Perbarui dukungan untuk masa depan Wilayah AWS](#)
- [Perbarui ketersediaan menurut negara](#)

Tambahkan sebuah Wilayah AWS

Anda dapat menambahkan Wilayah tempat pembeli dapat menggunakan produk Anda.

Untuk menambahkan Wilayah

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/>, lalu masuk ke akun penjual Anda.

2. Buka halaman [Produk Server](#), dan pada tab Produk server saat ini, pilih produk yang ingin Anda modifikasi.
3. Dari menu tarik-turun Permintaan perubahan, pilih Tambah Wilayah.
4. Pilih Wilayah yang ingin Anda tambahkan dari daftar Wilayah yang tersedia.
5. Pilih Kirim permintaan untuk mengirimkan permintaan Anda untuk ditinjau.
6. Verifikasi bahwa tab Permintaan menampilkan status Permintaan sebagai Dalam peninjauan. Ketika permintaan selesai, status menjadi Berhasil.

Note

Saat Anda menambahkan dukungan untuk yang baru Wilayah AWS, pelanggan yang sudah berlangganan penawaran pribadi untuk produk Anda tidak akan dapat mengakses Wilayah yang baru ditambahkan secara otomatis. Anda harus membuat penawaran pribadi lain dengan Wilayah yang ingin diakses pelanggan. Setelah menerima penawaran baru, pelanggan dapat mengakses Wilayah yang baru ditambahkan. Pelanggan yang berlangganan produk Anda di masa mendatang juga dapat mengakses Wilayah, selama Wilayah tersebut termasuk dalam penawaran pribadi. Untuk informasi selengkapnya tentang cara membuat penawaran pribadi baru, lihat [Peningkatan, pembaruan, dan perubahan penawaran pribadi](#).

Membatasi sebuah Wilayah AWS

Untuk mencegah pembeli baru menggunakan produk Anda secara spesifik Wilayah AWS, Anda dapat membatasi Wilayah. Anda dapat menambahkan Region kembali di lain waktu. Pelanggan produk yang ada di Wilayah dapat terus menggunakan produk dari Wilayah selama mereka berlangganan.

Untuk membatasi suatu Wilayah

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/>, lalu masuk ke akun penjual Anda.
2. Buka halaman [Produk Server](#), dan pada tab Produk server saat ini, pilih produk yang ingin Anda modifikasi.
3. Dari menu tarik-turun Permintaan perubahan, pilih Batasi Wilayah.
4. Pilih menu tarik-turun untuk melihat daftar Wilayah di mana produk Anda saat ini tersedia.

5. Pilih Wilayah yang ingin Anda batasi.
6. Wilayah yang Anda pilih muncul sebagai token. Tinjau daftar Wilayah yang Anda batasi, dan masukkan X untuk Wilayah yang tidak ingin Anda batasi.
7. Pilih Kirim permintaan perubahan untuk mengirimkan permintaan Anda untuk ditinjau.
8. Verifikasi bahwa tab Permintaan menampilkan status Permintaan sebagai Dalam peninjauan. Ketika permintaan selesai, status menjadi Berhasil.

Jika permintaan Anda berhasil, pengguna Anda yang ada akan menerima pesan email berikut yang memberi tahu mereka tentang Wilayah yang akan dibatasi. Mereka dapat terus menggunakan produk Anda selama mereka tetap berlangganan, tetapi mereka tidak dapat berlangganan ulang jika mereka membatalkan langganan.

Greetings from AWS Marketplace,

This message is a notification detailing a recent change for <ProductName>. {{{sellerName}}} has opted to restrict the <ProductType> product in <Restricted Region(s)> beginning <DateOfChange>.

This impacts you in the following ways:

1. As long as you're subscribed to the product, you can continue using the software product in the restricted Region.
2. You can't begin new instances of the software product in the restricted Region.
3. You can continue using the software product in all available AWS Regions.

Regards,

The AWS Marketplace Team

AWS, Inc. is a subsidiary of Amazon.com, Inc. Amazon.com (<http://amazon.com/>) is a registered

trademark of Amazon.com, Inc. This message was produced and distributed by Amazon Web Services Inc., 410 Terry Ave. North, Seattle, WA 98109-5210.

Perbarui dukungan untuk masa depan Wilayah AWS

Jika Anda ingin produk Anda di-onboard untuk yang baru diluncurkan Wilayah AWS, Anda dapat menggunakan dukungan Update future Region.

Untuk memperbarui dukungan Wilayah future

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/>, lalu masuk ke akun penjual Anda.
2. Buka halaman [Produk Server](#), dan pada tab Produk server saat ini, pilih produk yang ingin Anda modifikasi.
3. Dari menu tarik-turun Permintaan perubahan, pilih Perbarui dukungan Wilayah future.
4. Anda dapat memilih untuk mengaktifkan dukungan Wilayah future AWS Marketplace untuk memungkinkan onboard produk Anda yang baru diluncurkan Wilayah AWS atas nama Anda.
5. Setelah mengaktifkan fitur, Anda dapat memilih di antara semua Wilayah masa depan atau membatasi hanya untuk Wilayah AS.
6. Pilih Kirim permintaan perubahan untuk mengirimkan permintaan Anda untuk ditinjau.
7. Verifikasi bahwa tab Permintaan menampilkan status Permintaan sebagai Dalam peninjauan. Ketika permintaan selesai, status menjadi Berhasil.

Perbarui ketersediaan menurut negara

Jika Anda ingin mengubah negara tempat produk Anda dapat berlangganan dan ditawarkan, Anda dapat menggunakan Ketersediaan pembaruan.

Untuk memperbarui ketersediaan menurut negara

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/>, lalu masuk ke akun penjual Anda.
2. Buka halaman [Produk Server](#), pada tab Produk server saat ini, lalu pilih produk yang ingin Anda modifikasi.
3. Dari menu tarik-turun Permintaan perubahan, pilih Perbarui ketersediaan.
4. Pilih salah satu opsi berikut:
 1. Semua negara - Tersedia di semua negara yang didukung.
 2. Semua negara dengan pengecualian — Tersedia di semua negara yang didukung kecuali di negara tertentu.
 3. Daftar kustom - Daftar spesifik negara tempat produk tersedia.
5. Pilih Kirim permintaan perubahan untuk mengirimkan permintaan Anda untuk ditinjau.

6. Verifikasi bahwa tab Permintaan menampilkan status Permintaan sebagai Dalam peninjauan. Ketika permintaan selesai, status menjadi Berhasil.

Memperbarui perjanjian lisensi pengguna akhir (EULA) untuk produk EC2 Image Builder AWS Marketplace Anda

Sebagai AWS Marketplace penjual, Anda dapat memperbarui perjanjian lisensi pengguna akhir (EULA) yang akan mengatur penggunaan komponen EC2 Image Builder Anda. EULA Anda terletak di halaman daftar produk untuk daftar perangkat lunak publik di AWS Marketplace. Anda dapat menerapkan EULA Anda sendiri atau menggunakan [Kontrak Standar untuk AWS Marketplace \(SCMP\)](#). Prosedur berikut menunjukkan cara memperbarui EULA untuk komponen EC2 Image Builder Anda.

Untuk informasi lebih lanjut tentang EULA, lihat [Menggunakan kontrak standar di AWS Marketplace](#).

Untuk memperbarui EULA

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/>, lalu masuk ke akun penjual Anda.
2. Pilih tab [Produk Server](#), pada tab Produk server saat ini, pilih produk yang ingin Anda modifikasi.
3. Dari menu tarik-turun Permintaan perubahan, pilih Perbarui perjanjian lisensi pengguna akhir.
4. Anda dapat memilih [Kontrak Standar untuk AWS Marketplace \(SCMP\)](#) atau mengirimkan EULA kustom Anda sendiri. Untuk EULA kustom, Anda harus memberikan URL untuk kontrak kustom Anda dari bucket Amazon S3.

Note

Aksesibilitas publik harus diaktifkan di bucket Amazon S3 Anda.

5. Pilih Kirim permintaan perubahan untuk mengirimkan permintaan Anda untuk ditinjau.
6. Verifikasi bahwa tab Permintaan menampilkan status Permintaan sebagai Dalam peninjauan. Ketika permintaan selesai, status menjadi Berhasil.

Memperbarui kebijakan pengembalian dana untuk produk EC2 Image Builder Anda di AWS Marketplace

Sebagai AWS Marketplace penjual, Anda dapat mengatur kebijakan pengembalian dana untuk komponen EC2 Image Builder Anda. Jika Anda ingin mengubah kebijakan pengembalian dana untuk produk Anda, Anda dapat menggunakan kebijakan pengembalian dana Perbarui. Prosedur berikut menunjukkan cara memperbarui kebijakan pengembalian dana Anda.

Untuk memperbarui kebijakan pengembalian dana

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/>, lalu masuk ke akun penjual Anda.
2. Buka halaman [Produk Server](#), pada tab Produk server saat ini, lalu pilih produk yang ingin Anda modifikasi.
3. Dari menu tarik-turun Permintaan perubahan, pilih Perbarui kebijakan pengembalian dana.
4. Rincian kebijakan pengembalian dana saat ini disediakan di kotak teks untuk Anda edit. Mengirimkan permintaan menimpa kebijakan pengembalian dana saat ini.
5. Pilih Kirim permintaan perubahan untuk mengirimkan permintaan Anda untuk ditinjau.
6. Verifikasi bahwa tab Permintaan menampilkan status Permintaan sebagai Dalam peninjauan. Ketika permintaan selesai, status menjadi Berhasil.

Menghapus produk dari AWS Marketplace

Setelah Anda mempublikasikan produk Anda, Anda dapat menghapusnya AWS Marketplace. Ini juga disebut sebagai matahari terbenam. Untuk menghapus produk, identifikasi produk dan kirimkan permintaan untuk menghapusnya, bersama dengan alasan penghapusan dan alamat email kontak untuk Anda. Anda juga dapat memberikan ID produk pengganti jika Anda mengganti produk saat ini dengan yang baru.

Setelah Anda meminta penghapusan produk, pelanggan baru tidak dapat berlangganan produk. Anda harus mendukung pelanggan yang sudah ada selama minimal 90 hari.

Note

Anda tidak dapat menghapus produk yang dibatasi dari Portal AWS Marketplace Manajemen. Portal menyimpannya sebagai bagian dari riwayat publikasi akun.

Kami memproses permintaan penghapusan produk AWS Marketplace dengan ketentuan sebagai berikut:

- Produk dihapus dari AWS Marketplace pencarian, penelusuran, dan alat penemuan lainnya. Tombol atau fungsionalitas Berlangganan apa pun dinonaktifkan, dan pesan di halaman dengan jelas menunjukkan bahwa produk tidak lagi tersedia. Halaman detail produk masih dapat diakses menggunakan URL dan dapat diindeks di mesin pencari publik.
- Alasan penghapusan harus ditentukan (misalnya, akhir dukungan, akhir pembaruan produk, atau produk pengganti). Untuk persyaratan untuk melanjutkan dukungan untuk produk yang dihapus, lihat [Syarat dan Ketentuan untuk AWS Marketplace Penjual](#).
- AWS Marketplace menghubungi pembeli saat ini melalui pesan email yang memberi tahu mereka tentang penghapusan produk, alasan penghapusan, dan untuk memberikan informasi kontak penjual.
- Pembeli saat ini mempertahankan akses ke perangkat lunak hingga mereka membatalkan langganan mereka. Mereka tidak terpengaruh dengan cara apa pun oleh penghapusan produk.

Untuk menghapus produk yang dibuat menggunakan Portal Manajemen AWS Marketplace

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/>, lalu masuk ke akun penjual Anda.
2. Pilih **Produk** tab, kemudian pilih **Server**.
3. Pada halaman produk Anda, di bawah **Produk Server**, temukan produk yang ingin Anda hapus. Dari daftar tarik-turun **Permintaan perubahan**, pilih **Perbarui visibilitas produk**.
4. Pada halaman **Perbarui visibilitas produk**, pilih **Dibatasi**.
5. (Opsional) Sediakan ID Produk Penggantian, jika ada produk lain yang akan menggantikan produk yang Anda hapus.
6. Tinjau informasi untuk akurasi, lalu pilih **Kirim**.

Apa selanjutnya menampilkan halaman informasi setelah Anda mengirimkan permintaan penghapusan produk. Operasi AWS Marketplace Penjual meninjau dan memproses permintaan Anda. Periksa status kiriman Anda dengan melihat **Permintaan**.

Setelah produk Anda dihapus, produk akan muncul di daftar **Produk Saat Ini** di Portal Manajemen AWS Marketplace. Masuk **Produk** saat ini, satu-satunya tindakan yang dapat Anda lakukan adalah

mengunduh spreadsheet untuk produk tersebut. Anda tidak dapat mengedit atau mengirimkan permintaan matahari terbenam lainnya.

Jika Anda memiliki pertanyaan tentang penghapusan produk, hubungi [AWS Marketplace Tim Operasional Penjual](#).

Produk berbasis kontainer pada AWS Marketplace

AWS Marketplace mendukung produk perangkat lunak yang menggunakan wadah Docker. Produk kontainer terdiri dari opsi pengiriman yang merupakan sekumpulan gambar kontainer dan templat penyebaran yang berjalan bersamaan. Anda mengirimkan setidaknya satu opsi pengiriman untuk produk Anda, dengan maksimal empat. Untuk setiap opsi pengiriman, Anda menyediakan satu set gambar kontainer, petunjuk penggunaan, dan tautan ke templat penerapan bagi pelanggan untuk meluncurkan opsi pengiriman tersebut. Topik ini memberikan informasi tentang produk berbasis kontainer di AWS Marketplace

AWS Marketplace pembeli melihat opsi pengiriman yang tersedia di halaman detail produk yang dipublikasikan yang tersedia bagi mereka. Setelah mereka berlangganan produk dan memilih opsi pengiriman pilihan mereka, pembeli melihat informasi dan instruksi untuk meluncurkan dan menggunakan produk. Untuk opsi pengiriman gambar kontainer, pembeli melihat tautan ke templat penerapan dan gambar URLs kontainer yang tersedia. Mereka juga menerima instruksi tentang cara menarik gambar kontainer individu. Untuk opsi pengiriman bagan Helm, pembeli akan melihat step-by-step instruksi untuk meluncurkan menggunakan Helm.

Untuk panduan tentang pengalaman membeli, Anda dapat merujuk ke video ini: [Menyebarkan Kontainer AWS Marketplace di Amazon ECS Clusters](#) (3:34).

Anda dapat menemukan, berlangganan, dan menerapkan aplikasi Kubernetes pihak ketiga dari kluster Kubernetes mana pun AWS Marketplace di lingkungan apa pun. Anda dapat menerapkan aplikasi Kubernetes pihak ketiga di Amazon Elastic Container Service (Amazon ECS) Service (Amazon ECS), Amazon Elastic Kubernetes Service (Amazon EKS), dan lokal menggunakan Amazon EKS Anywhere (EKS AWS Fargate Anywhere). Anda juga dapat menerapkannya di kluster Kubernetes yang dikelola sendiri di tempat atau di Amazon Elastic Compute Cloud (Amazon). EC2

Anda dapat menjalankan produk kontainer Free and Bring Your Own License model (BYOL) pada runtime yang kompatibel dengan Docker.

Topik

- [Mendapatkan bantuan](#)
- [Memulai dengan Produk Kontainer](#)
- [Persyaratan produk berbasis kontainer untuk AWS Marketplace](#)
- [Harga produk kontainer untuk AWS Marketplace](#)
- [Integrasi penagihan, pengukuran, dan lisensi produk kontainer](#)

- [Pemberitahuan Amazon SNS untuk produk kontainer](#)

Mendapatkan bantuan

Untuk bantuan dengan produk kontainer Anda, hubungi mitra pengembangan bisnis Anda untuk AWS Marketplace atau tim [Operasi AWS Marketplace Penjual](#).

Memulai dengan Produk Kontainer

Sebagai AWS Marketplace penjual, Anda dapat membuat produk perangkat lunak berbasis kontainer. Produk kontainer terdiri dari opsi pengiriman yang merupakan sekumpulan gambar kontainer dan templat penyebaran yang berjalan bersamaan. Topik berikut menunjukkan kepada Anda bagaimana memulai dengan produk kontainer.

- [Siklus hidup produk](#)
- [Prasyarat](#)
- [Langkah 1: Buat ID produk dan kode produk untuk produk kontainer Anda](#)
- [Langkah 2: Buat daftar awal](#)
- [Langkah 3: Tambahkan versi awal produk Anda](#)
- [Langkah 4: \(Hanya untuk produk berbayar\) Integrasikan pengukuran atau harga kontrak](#)
- [Langkah selanjutnya](#)
- [Pemindaian produk kontainer untuk masalah keamanan](#)

Siklus hidup produk

Saat Anda membuat produk AWS Marketplace, produk tersebut awalnya diterbitkan dengan visibilitas terbatas sehingga akun di daftar izin dapat melihatnya, termasuk akun yang membuat produk. Ketika Anda siap, Anda dapat mempublikasikannya ke AWS Marketplace katalog untuk memungkinkan pembeli berlangganan dan membeli produk Anda.

Pada halaman [produk Server](#), Anda dapat melihat daftar produk Anda. Bergantung pada tahap apa, produk akan memiliki salah satu status berikut.

- Pementasan — Produk tidak lengkap yang masih Anda tambahkan informasinya. Pada Simpan pertama dan keluar dari pengalaman swalayan, permintaan perubahan yang berhasil membuat

produk yang tidak dipublikasikan dengan informasi dari langkah-langkah lengkap yang Anda kirimkan. Dari status ini, Anda dapat terus menambahkan informasi ke produk atau mengubah detail yang sudah dikirimkan melalui permintaan perubahan.

- **Terbatas** — Sebuah produk selesai setelah diserahkan ke sistem dan melewati semua validasi dalam sistem. Kemudian produk dirilis ke status Terbatas. Pada titik ini, produk memiliki halaman detail yang hanya dapat diakses oleh akun Anda dan siapa pun yang Anda izinkan. Anda dapat menguji produk Anda melalui halaman detail. Untuk informasi atau bantuan lebih lanjut, hubungi tim [Operasi AWS Marketplace Penjual](#).
- **Publik** — Saat Anda siap mempublikasikan produk sehingga pembeli dapat melihat dan berlangganan produk, Anda menggunakan permintaan perubahan visibilitas Perbarui. Permintaan ini memulai alur kerja untuk tim Operasi AWS Marketplace Penjual untuk meninjau dan mengaudit produk Anda terhadap AWS kebijakan. Setelah produk disetujui dan permintaan perubahan diproses, produk dipindahkan dari status Terbatas ke Publik. Untuk informasi tentang AWS pedoman, lihat [Persyaratan produk berbasis kontainer untuk AWS Marketplace](#).
- **Dibatasi** - Jika Anda ingin menghentikan pengguna baru berlangganan produk Anda, Anda dapat membatasi produk dengan menggunakan permintaan perubahan visibilitas Perbarui. Status Terbatas berarti bahwa pengguna yang terdaftar diizinkan yang ada dapat terus menggunakan produk. Namun, produk tidak akan lagi terlihat oleh publik atau tersedia untuk pengguna baru.

Prasyarat

Sebelum memulai, Anda harus menyelesaikan prasyarat berikut:

1. Akses dan gunakan [Portal Manajemen AWS Marketplace](#). Ini adalah alat yang Anda gunakan untuk mendaftar sebagai penjual dan mengelola produk yang Anda jual AWS Marketplace. Untuk informasi selengkapnya, lihat [Portal Manajemen AWS Marketplace](#).
2. Daftar sebagai penjual, dan kirimkan informasi pajak dan perbankan Anda. Untuk informasi selengkapnya, lihat [Mendaftar sebagai AWS Marketplace penjual](#).
3. Buat setidaknya satu kontainer di Amazon Elastic Container Service (Amazon ECS), Amazon Elastic Kubernetes Service (Amazon EKS), atau AWS Fargate Pastikan Anda memiliki tautan untuk gambar terkait.
4. Rencanakan bagaimana Anda akan membuat dan mengintegrasikan produk kontainer Anda AWS Marketplace.

Kami menyarankan Anda merencanakan strategi penetapan harga, hak, dan pengukuran Anda sebelum mempublikasikan produk Anda secara publik.

- Untuk informasi tentang persyaratan untuk produk berbasis kontainer, lihat [Persyaratan produk berbasis kontainer untuk AWS Marketplace](#).
- Untuk informasi tentang pengaturan harga untuk produk Anda, lihat [Harga produk kontainer untuk AWS Marketplace](#).
- Untuk informasi tentang pengukuran khusus untuk produk berbasis kontainer berbayar, lihat [Pengukuran per jam dan kustom dengan AWS Marketplace Metering Service](#)

Ikhtisar: Buat produk kontainer

Membuat Produk kontainer melibatkan langkah-langkah berikut:

1. [Langkah 1: Buat ID produk dan kode produk untuk produk kontainer Anda](#)
2. [Langkah 2: Buat daftar awal](#)
3. [Langkah 3: Tambahkan versi awal produk Anda](#)
4. [Langkah 4: \(Hanya untuk produk berbayar\) Integrasikan pengukuran atau harga kontrak](#)
5. [Perbarui visibilitas produk](#)

Untuk informasi tentang siklus hidup produk, lihat [Siklus hidup produk](#)

Langkah 1: Buat ID produk dan kode produk untuk produk kontainer Anda

Untuk memulai dengan produk kontainer, Anda harus membuat ID produk dan catatan kode produk AWS Marketplace. ID produk digunakan untuk melacak produk Anda sepanjang siklus hidupnya.

Gunakan prosedur berikut untuk membuat produk kontainer baru di Portal Manajemen AWS Marketplace, dan menghasilkan ID produk.

Note

Proses ini juga menciptakan kunci publik untuk wadah Anda yang berpasangan dengan produk Anda.

Untuk membuat ID produk kontainer

1. Buka browser web dan masuk ke file [Portal Manajemen AWS Marketplace](#).

2. Dari bilah menu, pilih Produk, dan pilih Server.
3. Pilih Buat produk server dan kemudian pilih Container.
4. Hasilkan ID dan kode produk kontainer.

 Note

(Opsional) Anda dapat menandai produk Anda untuk otorisasi berbasis tag. Untuk informasi selengkapnya, lihat [Menandai AWS sumber daya Anda](#).

5. Pilih Lanjutkan untuk terus membuat produk Anda.

Langkah 2: Buat daftar awal

Setelah membuat ID produk, kode produk, dan kunci publik, Anda akan menggunakan wizard untuk membuat daftar awal.

1. Berikan informasi produk untuk daftar produk Anda.
2. Tentukan model harga untuk produk Anda.

 Note

Untuk informasi lebih lanjut, lihat [Harga produk kontainer](#).

 Note

Untuk produk berbayar, produk Anda akan dimulai dengan harga \$0,01 untuk memungkinkan Anda dan tim Operasi AWS Marketplace Penjual untuk menguji produk tanpa menimbulkan biaya tinggi. Anda akan memberikan harga sebenarnya saat Anda go public.

3. Berikan informasi penawaran tambahan, termasuk kebijakan pengembalian dana, EULA, dan ketersediaan penawaran.
4. Tambahkan repositori awal untuk produk kontainer Anda.
5. Pilih Kirim pada langkah terakhir untuk memindahkan produk ke Visibilitas terbatas.

Note

Produk kontainer Anda awalnya dibuat dengan versi placeholder. Anda akan menambahkan versi final ketika produk memiliki visibilitas terbatas.

Langkah 3: Tambahkan versi awal produk Anda

Produk Anda mungkin memiliki beberapa versi selama masa pakainya. Setiap versi memiliki satu set citra kontainer yang khusus untuk versi tersebut. Untuk menambahkan versi awal produk Anda, lihat [Menambahkan versi baru produk kontainer Anda di AWS Marketplace](#).

Langkah 4: (Hanya untuk produk berbayar) Integrasikan pengukuran atau harga kontrak

Untuk produk berbasis kontainer dengan harga penggunaan, Anda menggunakan [AWS Marketplace Metering Service](#) untuk memeriksa hak untuk menggunakan produk Anda dan penggunaan pengukuran untuk penagihan. Anda harus mengukur model harga yang Anda buat saat mengatur informasi harga Anda. Untuk informasi selengkapnya, lihat [Pengukuran per jam dan kustom dengan AWS Marketplace Metering Service](#).

Harga kontrak

Untuk produk berbasis kontainer dengan harga kontrak, Anda menggunakan lisensi AWS License Manager untuk mengaitkan lisensi dengan produk Anda.

Untuk informasi lebih lanjut tentang mengintegrasikan dengan AWS License Manager, lihat [Harga kontrak untuk produk kontainer dengan AWS License Manager](#).

Langkah 5: Perbarui visibilitas produk

Saat Anda membuat produk AWS Marketplace, produk tersebut awalnya diterbitkan dengan visibilitas terbatas sehingga akun di daftar izin dapat melihatnya, termasuk akun yang membuat produk. Anda dapat memperbarui visibilitas produk untuk memungkinkan pembeli berlangganan dan membeli produk Anda. Bergantian Anda dapat memperbarui daftar izinkan produk untuk ditambahkan. Akun AWS Topik ini menunjukkan kepada Anda bagaimana mengelola pembeli mana yang dapat melihat produk Anda AWS Marketplace.

Untuk informasi selengkapnya tentang visibilitas produk dan siklus hidup, lihat [Siklus hidup produk](#)

Topik

- [Perbarui visibilitas produk](#)
- [Memperbarui daftar yang diizinkan Akun AWS IDs](#)

Perbarui visibilitas produk

Untuk memperbarui visibilitas

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/>, lalu masuk ke akun penjual Anda.
2. Buka halaman [Produk Server](#), pada tab Produk server saat ini, pilih produk berbasis kontainer yang ingin Anda modifikasi.
3. Dari menu tarik-turun Permintaan perubahan, pilih Perbarui visibilitas.

Note

Anda dapat meminta agar produk dipindahkan dari status Terbatas ke status Publik dengan menggunakan permintaan perubahan ini. Namun, permintaan perubahan harus melalui proses persetujuan tim Operasi AWS Marketplace Penjual untuk dipindahkan ke Publik.

4. Pilih Kirim untuk mengirimkan permintaan Anda untuk ditinjau.
5. Verifikasi bahwa tab Permintaan menampilkan status Permintaan sebagai Dalam peninjauan. Ketika permintaan selesai, status menjadi Berhasil.

Memperbarui daftar yang diizinkan Akun AWS IDs

Anda dapat mengubah daftar Akun AWS IDs yang dapat melihat produk Anda dalam keadaan terbatas. Akun yang terdaftar diizinkan menampilkan lencana Terbatas di samping versi produk pada halaman detail produk.

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/> dan masuk ke akun penjual Anda.
2. Dari halaman [Produk Server](#), pilih produk kontainer yang ingin Anda modifikasi.
3. Dari daftar tarik-turun Permintaan perubahan, pilih Perbarui daftar yang diizinkan. Daftar akun saat ini yang diizinkan ditampilkan.

4. Di Akun AWS bidang Allowlisted, masukkan Akun AWS IDs dan pisahkan menggunakan koma.
5. Pilih Kirim untuk mengirimkan permintaan Anda untuk ditinjau.
6. Verifikasi bahwa tab Permintaan menampilkan status Permintaan sebagai Dalam peninjauan. Ketika permintaan selesai, status akan diperbarui ke Berhasil atau Gagal.

Langkah selanjutnya

Setelah membuat produk kontainer, Anda dapat menggunakan informasi dalam topik berikut untuk mengonfigurasi dan mengelolanya:

- [Memperbarui informasi produk untuk produk kontainer Anda di AWS Marketplace](#)
- [Menambahkan versi baru produk kontainer Anda di AWS Marketplace](#)
- [Mengelola harga produk kontainer AWS Marketplace](#)
- [Memperbarui ketersediaan produk kontainer menurut negara di AWS Marketplace](#)
- [Memperbarui perjanjian lisensi pengguna akhir \(EULA\) Anda untuk produk kontainer AWS Marketplace](#)
- [Menguji dan melepaskan produk kontainer Anda AWS Marketplace](#)

Pemindaian produk kontainer untuk masalah keamanan

Saat Anda membuat permintaan perubahan untuk menambahkan versi baru ke produk kontainer Anda, kami memindai gambar kontainer yang disertakan dalam versi baru tersebut dan memeriksa kerentanan keamanan. Untuk melakukan ini, kami melakukan pemindaian layer-by-layer statis pada gambar. Jika kami menemukan kerentanan kritis dengan vektor risiko yang dapat dieksploitasi dari jarak jauh, kami menyediakan daftar masalah yang ditemukan. Kami sangat menyarankan agar Anda melakukan analisis keamanan Anda sendiri menggunakan pemindai gambar kontainer seperti Clair, Twistlock, Aqua Security, atau Trend Micro untuk menghindari keterlambatan dalam proses konsumsi dan penerbitan.

Pilihan citra dasar Anda untuk membangun citra kontainer Anda dapat memiliki pengaruh yang signifikan pada profil keamanan citra akhir. Jika Anda memilih citra dasar yang sudah diketahui kerentanan kritis, mereka akan ditAndai karena lapisan dasar, bahkan jika lapisan perangkat lunak aplikasi Anda bersih. Kami menyarankan Anda memverifikasi bahwa Anda memulai dengan wadah dasar yang bebas dari kerentanan sebelum Anda membuat gambar Anda dan mengirimkannya ke AWS Marketplace.

Memperbarui informasi produk untuk produk kontainer Anda di AWS Marketplace

Setelah Anda membuat ID produk dan menetapkan harga, Anda dapat mengedit informasi produk Anda, termasuk apa yang akan dilihat pelanggan tentang produk kontainer Anda di AWS Marketplace. Misalnya, jika versi baru memodifikasi deskripsi atau sorotan produk, Anda dapat mengedit informasi produk dengan data baru. Anda juga dapat memperbarui informasi produk lainnya, termasuk judul produk, SKU deskripsi, kategori, kata kunci, dan lainnya. Prosedur berikut menguraikan membuat detail produk untuk produk Anda.

Untuk membuat atau memperbarui detail produk untuk produk kontainer Anda

1. Masuk ke [Portal Manajemen AWS Marketplace](#).
2. Pilih Server dari menu Produk.
3. Pada tab Produk Server, pilih produk yang ingin Anda modifikasi.
4. Dari menu menurun Minta perubahan, pilih Perbarui informasi produk.
5. Perbarui salah satu bidang berikut yang ingin Anda ubah:
 - Judul produk
 - SKU
 - Deskripsi singkat
 - Deskripsi panjang
 - Gambar logo produk URL
 - Sorotan
 - Kategori produk
 - Kata kunci
 - Video produk URL
 - Sumber Daya
 - Informasi Support

Note

Gambar URLs harus berada dalam bucket Amazon S3 yang dapat diakses publik. Untuk detail lebih lanjut tentang format logo, lihat [Persyaratan logo perusahaan dan produk](#).

6. Pilih Kirim.
7. Verifikasi bahwa permintaan muncul di tab Permintaan dengan status Dalam tinjauan. Anda mungkin perlu me-refresh halaman untuk melihat permintaan pada daftar.

Anda dapat memeriksa status permintaan Anda kapan saja dari tab Permintaan halaman [Produk server](#).

Menambahkan versi baru produk kontainer Anda di AWS Marketplace

Sebagai AWS Marketplace penjual, Anda dapat menambahkan versi baru produk kontainer, mengelola versi, dan memperbarui informasi versi. Produk Anda mungkin memiliki beberapa versi selama masa pakainya. Setiap versi memiliki satu set citra kontainer yang khusus untuk versi tersebut. Topik berikut menunjukkan cara mengelola versi produk untuk produk kontainer Anda.

Note

Anda tidak dapat menambahkan versi ke produk Anda sampai Anda telah membuat ID produk dan harga untuk produk Anda. Untuk informasi lebih lanjut tentang langkah-langkah tersebut, lihat [Langkah 1: Buat ID produk dan kode produk untuk produk kontainer Anda](#).

Topik

- [Langkah 1: Tambahkan repositori](#)
- [Langkah 2: Unggah gambar kontainer dan artefak ke repositori](#)
- [Langkah 3: Tambahkan versi baru ke produk kontainer Anda](#)
- [Langkah 4: Perbarui informasi versi](#)
- [Batasi versi add-on Amazon EKS Anda](#)

Langkah 1: Tambahkan repositori

Citra kontainer dan artefak lainnya untuk produk Anda disimpan dalam repositori di AWS Marketplace. Biasanya, Anda membuat satu repositori untuk setiap artefak yang dibutuhkan, tetapi repositori dapat menyimpan beberapa versi artefak (dengan tag yang berbeda).

Note

Semua citra dalam deployment produk Anda harus menggunakan citra dari repositori AWS Marketplace .

Prosedur berikut menjelaskan cara menambahkan repositori yang diperlukan di. AWS Marketplace

Untuk menambahkan repositori

1. Masuk ke [Portal Manajemen AWS Marketplace](#).
2. Pilih Server dari menu Produk.
3. Pada tab Produk Server, pilih produk yang ingin Anda ubah, lalu pilih Tambahkan repositori dari menu tarik-turun Permintaan perubahan.
4. Masukkan nama repositori yang ingin Anda buat. Jika Anda ingin membuat lebih dari satu repositori baru:
 - Pilih Tambahkan repositori baru untuk setiap repositori tambahan.
 - Berikan nama yang unik. Nama unik yang Anda pilih harus ada di semua produk di akun penjual Anda.

Note

Repositori akan memiliki struktur ini: `<repositoryID>.dkr.ecr.us-east-1.amazonaws.com/<sellerName>/<repositoryName>` Ketika Anda menambahkan item ke repositori (dalam prosedur berikut), mereka akan mendapatkan tag dan memiliki struktur ini: `<repositoryID>.dkr.ecr.us-east-1.amazonaws.com/<sellerName>/<repositoryName>:<tag>` repositoryID ini adalah ID internal untuk AWS Marketplace. sellerName ini didasarkan pada nama yang Anda buat untuk akun penjual Anda. Anda mendefinisikan

`repositoryName` dalam langkah ini. tag ini diatur saat Anda mengunggah artefak ke repositori.

5. Pilih Kirim.

Note

Anda dapat memiliki hingga 50 repositori per produk.

Permintaan baru dibuat dan ditampilkan pada tab Permintaan. Setelah selesai, dalam beberapa menit, Anda dapat mulai menambahkan gambar kontainer dan artefak lainnya ke repositori yang telah Anda buat.

Langkah 2: Unggah gambar kontainer dan artefak ke repositori

Untuk mengunggah citra kontainer dan artefak ke repositori

1. Masuk ke [Portal Manajemen AWS Marketplace](#).
2. Dari menu Produk, pilih Server.
3. Pada tab Produk Server, pilih produk yang ingin Anda modifikasi.
4. Pilih Tambahkan repositori dari menu menurun Minta perubahan.
5. Pilih Lihat repositori yang ada.
6. Pilih repositori yang ingin Anda unggah.
7. Pilih Lihat perintah push untuk membuka daftar instruksi, termasuk perintah yang dapat Anda gunakan untuk mendorong Docker gambar kontainer dan Helm grafik ke repositori itu.

Untuk informasi umum tentang cara mendorong gambar kontainer dan artefak lainnya ke repositori, lihat [Mendorong gambar](#) di Panduan Pengguna Amazon Elastic Container Registry.

Note

Anda dapat menggunakan operasi API Amazon Elastic Container Registry (Amazon ECR) Registry ECR) berikut saat memanggil atau: `docker pull` `docker push`

- `DescribeImages`— Gunakan ini untuk meninjau metadata tentang gambar dalam repositori.

- `GetAuthorizationToken`— Gunakan untuk mengautentikasi sebelum mengunggah artefak ke repositori, lalu gunakan atau perintah. `docker pull` `docker push`
- `ListImages`— Gunakan untuk melihat daftar gambar yang Anda dorong.

8. Gunakan perintah yang terdaftar untuk mendorong artefak yang diperlukan dari repositori lokal Anda ke repositori untuk produk AWS Marketplace Anda.

 Note

tag Yang Anda berikan dalam push perintah digunakan untuk membedakan versi artefak yang Anda unggah ke repositori. Gunakan tag yang masuk akal untuk versi artefak adalah bagian dari.

9. Ulangi untuk setiap citra kontainer atau artefak yang Anda butuhkan dalam versi Anda.

 Note

Versi Anda dapat menyertakan hingga 50 gambar kontainer atau artefak di setiap opsi pengiriman. Lihat prosedur berikut untuk informasi lebih lanjut tentang opsi pengiriman.

Setelah Anda mengunggah artefak Anda, Anda siap untuk membuat versi produk Anda.

 Note

Citra kontainer Anda dipindai secara otomatis untuk melihat apakah citra tersebut memenuhi [Persyaratan produk berbasis kontainer untuk AWS Marketplace](#). Untuk informasi lebih lanjut, lihat [Pemindaian produk kontainer untuk masalah keamanan](#).

Menambahkan opsi pengiriman baru

Setiap versi produk kontainer Anda akan membutuhkan opsi pengiriman. Opsi pengiriman menentukan opsi penyebaran yang tersedia untuk pembeli. Bergantung pada salah satu opsi pengiriman di bawah ini, Anda perlu mengunggah artefak yang sesuai ke dalam repositori.

- Untuk opsi pengiriman gambar Container, unggah semua gambar kontainer yang diperlukan untuk instalasi produk ke dalam repositori Amazon Elastic Container Registry (Amazon ECR) yang dibuat di konsol. AWS Marketplace
- Untuk Helm opsi pengiriman bagan, unggah Helm grafik dan gambar kontainer ke dalam repositori Amazon ECR yang dibuat di konsol. AWS Marketplace
- Untuk opsi pengiriman add-on konsol Amazon EKS, unggah Helm grafik dan gambar kontainer ke dalam repositori Amazon ECR yang dibuat di konsol. AWS Marketplace

Langkah 3: Tambahkan versi baru ke produk kontainer Anda

Note

Jika Anda menerima kesalahan saat menambahkan versi baru ke penampung, lihat [tabel Tambahkan Kesalahan Asinkron versi baru](#) di Referensi.AWS Marketplace Catalog API

Cara menambahkan versi baru ke produk kontainer

1. Masuk ke [Portal Manajemen AWS Marketplace](#).
2. Pilih Server dari menu Produk.
3. Pada tab Produk Server, pilih produk yang ingin Anda tambahkan versinya. Kemudian pilih Tambahkan versi baru dari dropdown Permintaan perubahan.
4. Pada halaman Tambahkan versi baru, masukkan Judul versi dan Catatan rilis untuk versi Anda.
5. Setelah memasukkan detail versi, langkah selanjutnya adalah menambahkan opsi pengiriman. Opsi pengiriman adalah serangkaian instruksi dan informasi yang dapat digunakan pembeli untuk meluncurkan perangkat lunak dari versi produk Anda. Opsi pengiriman dikenal sebagai opsi pemenuhan kepada pembeli.

Note

Produk Anda dapat mendukung beberapa platform dengan gambar kontainer yang berbeda (misalnya, Kubernetes and Ubuntu penyebaran). Anda dapat membuat satu opsi pengiriman untuk setiap cara pelanggan dapat mengatur produk Anda, hingga empat opsi pengiriman per versi produk.

- a. Jika produk sudah memiliki opsi pengiriman di versi lain, Anda dapat menggunakan opsi yang ada sebagai templat untuk menambahkan opsi pengiriman ke versi baru. Di opsi Pengiriman, pilih opsi pengiriman yang ingin Anda tambahkan dari daftar. Anda dapat mengedit opsi menggunakan instruksi dalam langkah-langkah berikut.
 - b. Untuk menambahkan opsi pengiriman baru, pilih Opsi pengiriman baru. Setelah menambahkan opsi, ikuti instruksi dalam langkah-langkah berikut untuk mengkonfigurasinya.
6. Pilih metode pengiriman untuk opsi pengiriman. Metode pengiriman menentukan bagaimana pembeli akan meluncurkan perangkat lunak Anda.
- Untuk opsi pengiriman gambar Container, berikan jalur ke gambar kontainer di repositori Amazon Elastic Container Registry (Amazon ECR) yang dibuat di konsol. AWS Marketplace Pembeli menggunakan jalur gambar kontainer untuk meluncurkan perangkat lunak dengan menarik gambar langsung ke lingkungan mereka.
 - Untuk Helm opsi pengiriman bagan, berikan jalur ke Helm grafik di repositori Amazon ECR yang dibuat di konsol. AWS Marketplace Pembeli menginstal Helm grafik di lingkungan penyebaran mereka untuk meluncurkan perangkat lunak.
 - Untuk opsi pengiriman add-on konsol Amazon EKS, berikan jalur ke Helm grafik di repositori Amazon ECR yang dibuat di konsol. AWS Marketplace Pembeli menginstal wadah menggunakan konsol Amazon EKS atau add-on Amazon EKS asli APIs untuk meluncurkan perangkat lunak. Untuk informasi selengkapnya, lihat [Add-on Amazon EKS yang tersedia dari Amazon EKS](#).
- a. Untuk menambahkan opsi pengiriman gambar Container, lakukan langkah-langkah berikut:
 - i. Dalam gambar Container, tambahkan URL Amazon ECR ke gambar kontainer yang berisi perangkat lunak versi produk.
 - ii. Dalam judul opsi Pengiriman dan deskripsi opsi Deployment, masukkan judul dan deskripsi untuk opsi pengiriman ini.
 - iii. Dalam petunjuk Penggunaan, masukkan informasi terperinci untuk membantu pembeli Anda menggunakan perangkat lunak Anda setelah meluncurkannya.
 - iv. Di Layanan yang didukung, pilih lingkungan tempat pembeli dapat meluncurkan perangkat lunak.

- v. Dalam template Deployment, tambahkan sumber daya yang dapat digunakan pembeli untuk meluncurkan perangkat lunak. Masukkan judul dan URL ke sumber daya untuk setiap template.
- b. Untuk menambahkan Helm opsi pengiriman bagan, lakukan langkah-langkah berikut:
 - i. Di Helm bagan, tambahkan URL Amazon ECR ke Helm bagan bahwa pembeli akan menginstal di lingkungan penyebaran mereka untuk meluncurkan perangkat lunak Anda.
 - ii. Dalam gambar Container, tambahkan URL Amazon ECR ke gambar kontainer yang berisi perangkat lunak versi produk.
 - iii. Dalam judul opsi Pengiriman dan deskripsi opsi Deployment, masukkan judul dan deskripsi untuk opsi pengiriman ini.
 - iv. Dalam petunjuk Penggunaan, masukkan informasi terperinci untuk membantu pembeli Anda menggunakan perangkat lunak Anda setelah meluncurkannya.
 - v. Di Layanan yang didukung, pilih lingkungan tempat pembeli dapat meluncurkan perangkat lunak.
 - vi. Opsional - Dalam Helm nama rilis, masukkan nama Kubernetes namespace di mana Helm grafik akan diinstal.
 - vii. Opsional - Dalam Helm namespace instalasi, masukkan nama untuk Helm rilis yang akan digunakan oleh `helm install` perintah.
 - viii. Opsional - Dalam Kubernetes nama akun layanan, masukkan nama Kubernetes akun layanan yang akan digunakan untuk terhubung ke AWS Identity and Access Management (IAM). Bagian Kubernetes layanan panggilan akun AWS layanan seperti lisensi atau pengukuran.
 - ix. Pilih untuk mengaktifkan QuickLaunch pada versi produk ini. Quick Launch adalah fitur di AWS Marketplace. Pembeli dapat menggunakan Quick Launch untuk membuat cluster Amazon EKS dengan cepat dan meluncurkan perangkat lunak Anda di dalamnya dengan menggunakan AWS CloudFormation. Untuk informasi selengkapnya, lihat [Peluncuran Cepat di AWS Marketplace](#).
 - x. Dalam parameter Override, masukkan parameter yang akan digunakan di Helm Perintah CLI yang meluncurkan perangkat lunak. Parameter ini memungkinkan pembeli untuk mengganti nilai default yang disediakan. Jika Anda telah mengaktifkan Quick Launch, masukkan juga nama parameter dan deskripsi untuk CloudFormation formulir. Ada batas 15 parameter saat menggunakan Konsol AWS Marketplace Manajemen,

tetapi tidak ada batasan saat menggunakan AWS Marketplace Catalog API. Untuk informasi selengkapnya, lihat [Menambahkan versi baru ke produk berbasis container](#).

 Note

Beberapa parameter Override diperlukan. Produk Amazon EKS Anywhere memerlukan parameter Override untuk rahasia lisensi dengan DefaultValue dari. "\${AWSMP_LICENSE_SECRET}" Untuk produk berbayar, Anda harus memberikan satu parameter Override untuk konfigurasi akun layanan dengan DefaultValue dari. "\${AWSMP_SERVICE_ACCOUNT}"

- xi. Pilih Sembunyikan kata sandi dan rahasia untuk menutupi informasi sensitif di konsol, alat baris perintah, dan APIs. Untuk informasi selengkapnya, lihat dokumentasi NoEcho [parameter di Parameter](#) di Panduan AWS CloudFormation Pengguna.
- c. Untuk menambahkan opsi pengiriman add-on konsol Amazon EKS, pastikan artefak sesuai [Persyaratan untuk produk add-on Amazon EKS](#), lalu lakukan langkah-langkah berikut:

 Note

Hanya satu opsi pengiriman add-on Amazon EKS yang didukung per versi. Anda tidak dapat menambahkan versi baru hingga versi saat ini yang Anda kerjakan dipublikasikan di konsol Amazon EKS.

- i. Di Helm bagan, tambahkan URL Amazon ECR ke Helm bagan bahwa pembeli akan menginstal di lingkungan penyebaran mereka untuk meluncurkan perangkat lunak Anda.
- ii. Dalam gambar Container, tambahkan URL Amazon ECR ke gambar kontainer yang berisi perangkat lunak versi produk. Pastikan bahwa semua gambar di dalam Helm bagan terdaftar.
- iii. Dalam judul opsi Pengiriman dan deskripsi opsi Deployment, masukkan judul dan deskripsi untuk opsi pengiriman ini.
- iv. Dalam Visibilitas, pertahankan nilai default Limited dipilih.
- v. Di Add-on name, masukkan nama unik untuk add-on ini. Nama add-on yang Anda masukkan akan ditambahkan dengan nama penjual saat ditampilkan di konsol Amazon EKS.

- vi. Dalam versi Add-on, masukkan versi add-on yang akan terlihat saat menginstal atau memutakhirkan add-on ini. Ikuti formatnya `major.minor.patch`.
 - vii. Pada tipe Add-on, pilih kategori untuk add-on Anda dari daftar dropdown.
 - viii. Di Kubernetes Versi, pilih semua Kubernetes versi yang akan didukung add-on Anda.
 - ix. Di Arsitektur, pilih arsitektur platform yang didukung add-on Anda. Opsi nya adalah AMD64 dan ARM64. Kami merekomendasikan mendukung kedua arsitektur untuk memaksimalkan kompatibilitas. Jika add-on Anda tidak mendukung ARM64 perangkat, Anda harus menentukan tanggal yang direncanakan untuk menambahkan dukungan sebelum produk Anda dapat dipublikasikan di semua iklan Wilayah AWS.
 - x. Di Namespace, masukkan yang unik Kubernetes namespace tempat add-on Anda akan diinstal. `kube-public` Ruang nama default `kube-system`, dan tidak didukung untuk menginstal add-on pihak ketiga.
 - xi. Dalam parameter Environment Override, Anda dapat memilih hingga 2 parameter lingkungan dari kerangka kerja add-on Amazon EKS. Anda dapat memetakan nama parameter dari `values.yaml` Anda ke variabel lingkungan ini, yaitu `dan`.
`${AWS_REGION} ${AWS_EKS_CLUSTER_NAME}`
7. Untuk menambahkan opsi pengiriman tambahan, pilih opsi Pengiriman baru dan ulangi instruksi di langkah sebelumnya untuk mengonfigurasinya.
 8. Pilih Kirim.

Langkah 4: Perbarui informasi versi

Setelah versi dibuat, dapat membantu untuk memberikan informasi terbaru kepada pembeli Anda dengan memodifikasi informasi yang terkait dengan versi. Misalnya, jika Anda berencana untuk membatasi versi 1.0 setelah versi 1.1 dirilis, Anda dapat memperbarui deskripsi versi 1.0 untuk mengarahkan pembeli ke versi 1.1. Berikan tanggal bahwa versi 1.0 akan dibatasi. Anda memperbarui informasi versi dari file Portal Manajemen AWS Marketplace.

Untuk memperbarui informasi versi

1. Masuk ke [Portal Manajemen AWS Marketplace](#).
2. Pilih Server dari menu Produk.
3. Pada tab Produk Server, pilih produk yang ingin Anda modifikasi.
4. Dari menu menurun Minta perubahan, pilih Perbarui informasi versi.
5. Pada halaman Perbarui versi, pilih versi yang ingin Anda perbarui.

6. Buat pembaruan ke versi yang dipilih. Bidang yang tersedia untuk diperbarui bergantung pada status versi produk atau opsi pengiriman.
 - a. Untuk semua versi, Anda dapat memperbarui catatan Rilis.
 - b. Untuk versi yang belum tersedia untuk umum, Anda dapat memperbarui judul Versi.
 - c. Untuk opsi pengiriman yang belum dibatasi, Anda dapat memperbarui bidang berikut:
 - Deskripsi
 - Instruksi penggunaan
 - Layanan yang didukung
 - d. Untuk opsi pengiriman dalam versi yang belum tersedia untuk umum, Anda dapat memperbarui bidang berikut:
 - Judul opsi pengiriman
 - Helm grafik (untuk Helm opsi pengiriman bagan saja)
 - Gambar kontainer
 - Sumber daya penyebaran
 - AddOn Nama
 - AddOn Versi
 - AddOn Jenis
 - Helm Grafik URI
 - CompatibleKubernetesVersions
 - SupportedArchitectures
 - Namespace
 - EnvironmentOverrideParameters
 - e. Untuk opsi pengiriman dalam versi yang tersedia untuk umum, Anda dapat memperbarui SupportedArchitectures.
7. Pilih Kirim.
8. Verifikasi bahwa permintaan muncul di tab Permintaan dengan status Dalam tinjauan.

Anda dapat memeriksa status permintaan Anda kapan saja dari tab Permintaan halaman [Produk server](#).

Batasi versi add-on Amazon EKS Anda

Untuk membatasi versi produk kontainer Anda yang diterbitkan sebagai add-on, hubungi tim AWS Marketplace operasi menggunakan formulir hubungi kami di bagian bawah. [Portal Manajemen AWS Marketplace](#)

Mengelola harga produk kontainer AWS Marketplace

Sebagai AWS Marketplace penjual, Anda dapat mengelola dan memperbarui harga produk kontainer. Untuk produk kontainer, Anda dapat membuat daftar produk gratis, produk model Bring Your Own License (BYOL), dan produk berbayar untuk Amazon Elastic Container Service (AmazonECS), Amazon Elastic Kubernetes Service (EKSAmazon), dan AWS Fargate Anda hanya dapat menetapkan satu harga per produk. Anda dapat menambah atau memperbarui dimensi harga. Bergantung pada model harga Anda, Anda dapat menambahkan dimensi kontrak dan penggunaan. Anda juga dapat memperbarui ketentuan harga atau kebijakan pengembalian dana Anda. Untuk informasi selengkapnya, silakan lihat bagian-bagian berikut ini.

Untuk informasi selengkapnya tentang model penetapan harga produk kontainer AWS Marketplace, lihat [Harga produk kontainer untuk AWS Marketplace](#).

Topik

- [Menambahkan dimensi harga](#)
- [Memperbarui informasi dimensi](#)
- [Memperbarui ketentuan harga](#)
- [Memperbarui kebijakan pengembalian dana suatu produk](#)

Menambahkan dimensi harga

Anda dapat menambahkan dimensi harga ke model penetapan harga yang Anda gunakan untuk penagihan produk Anda. Untuk informasi selengkapnya tentang model penetapan harga, lihat [Model harga kontainer](#).

Note

Menambahkan dimensi harga untuk kontrak dengan harga konsumsi (misalnya, pay-as-you-go penetapan harga untuk penggunaan tambahan) tidak tersedia di Portal Manajemen AWS Marketplace.

Anda tidak dapat mengubah model harga antara kontrak, penggunaan, dan kontrak dengan harga konsumsi. Hubungi tim [Operasi AWS Marketplace Penjual](#) untuk bantuan.

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/> dan masuk ke akun penjual Anda.
2. Dari halaman [Produk Server](#), pilih produk kontainer yang ingin Anda modifikasi.
3. Dari daftar tarik-turun Permintaan perubahan, pilih Perbarui dimensi harga, lalu pilih Tambahkan dimensi harga.
4. Bergantung pada model harga, Anda dapat menambahkan dimensi kontrak atau dimensi penggunaan dengan memberikan informasi untuk API pengenal, Nama tampilan, dan Deskripsi.
5. Pilih Berikutnya, dan masukkan harga dimensi kontrak Anda.
6. Pilih Kirim untuk mengirimkan permintaan Anda untuk ditinjau.
7. Verifikasi bahwa tab Permintaan menampilkan status Permintaan sebagai Dalam peninjauan. Ketika permintaan selesai, status akan diperbarui ke Berhasil atau Gagal.

Memperbarui informasi dimensi

Anda dapat mengubah informasi dimensi untuk produk Anda. Untuk informasi selengkapnya tentang model penetapan harga, lihat [Model harga kontainer](#).

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/> dan masuk ke akun penjual Anda.
2. Dari halaman [Produk Server](#), pilih produk kontainer yang ingin Anda modifikasi.
3. Dari daftar tarik-turun Permintaan perubahan, pilih Perbarui dimensi harga, lalu pilih Perbarui informasi dimensi.
4. Anda dapat menambahkan informasi dimensi dengan memberikan informasi untuk nama Tampilan dan Deskripsi dimensi yang ingin Anda perbarui.
5. Pilih Kirim untuk mengirimkan permintaan Anda untuk ditinjau.
6. Verifikasi bahwa tab Permintaan menampilkan status Permintaan sebagai Dalam peninjauan. Ketika permintaan selesai, status akan diperbarui ke Berhasil atau Gagal.

Memperbarui ketentuan harga

Anda dapat mengubah ketentuan harga untuk produk Anda. Ketentuan harga mungkin perlu diperbarui jika Anda mengubah negara tempat produk Anda ditawarkan.

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/> dan masuk ke akun penjual Anda.
2. Dari halaman [Produk Server](#), pilih produk kontainer yang ingin Anda modifikasi.
3. Dari daftar Permintaan perubahan, pilih Perbarui penawaran umum, lalu pilih Perbarui persyaratan harga.
4. Harga saat ini sudah diisi sebelumnya untuk Anda edit. Anda dapat menghapus harga saat ini dan menambahkan harga baru Anda. Kami menyarankan Anda meninjau harga yang Anda minta sebelum mengirimkan permintaan Anda untuk ditinjau.
5. Pilih Kirim untuk mengirimkan permintaan Anda untuk ditinjau.
6. Verifikasi bahwa tab Permintaan menampilkan status Permintaan sebagai Dalam peninjauan. Ketika permintaan selesai, status akan diperbarui ke Berhasil atau Gagal.

Note

Jika Anda menaikkan harga dimensi menggunakan ketentuan harga Pembaruan, Anda tidak akan dapat memperbarui harga selama 90 hari. Kenaikan harga mengunci harga selama 90 hari sejak hari Anda memulai perubahan. Kunci harga ini hanya berlaku untuk kenaikan harga bukan penurunan harga.

Memperbarui kebijakan pengembalian dana suatu produk

Anda dapat memperbarui kebijakan pengembalian dana untuk produk Anda. Pembaruan kebijakan pengembalian dana berlaku untuk semua pengguna. Untuk informasi selengkapnya, lihat [Pengembalian uang produk di AWS Marketplace](#).

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/> dan masuk ke akun penjual Anda.
2. Dari halaman [Produk Server](#), pilih produk kontainer yang ingin Anda modifikasi.
3. Dari daftar tarik-turun Permintaan perubahan, pilih Perbarui penawaran umum, lalu pilih Perbarui kebijakan pengembalian dana.

4. Kebijakan pengembalian dana saat ini telah diisi sebelumnya untuk Anda edit. Anda dapat menghapus kebijakan pengembalian dana saat ini dan menambahkan kebijakan pengembalian dana baru Anda. Kami menyarankan Anda meninjau kebijakan pengembalian dana yang Anda minta sebelum mengirimkan permintaan Anda untuk ditinjau. Mengirimkan permintaan menimpa kebijakan pengembalian dana saat ini.
5. Pilih Kirim untuk mengirimkan permintaan Anda untuk ditinjau.
6. Verifikasi bahwa tab Permintaan menampilkan status Permintaan sebagai Dalam peninjauan. Ketika permintaan selesai, status akan diperbarui ke Berhasil atau Gagal.

Memperbarui ketersediaan produk kontainer menurut negara di AWS Marketplace

Sebagai AWS Marketplace penjual, Anda dapat mengubah negara tempat produk kontainer Anda dapat ditawarkan dan berlangganan. Untuk informasi selengkapnya, lihat [Negara-negara](#). Prosedur berikut menunjukkan kepada Anda cara memperbarui ketersediaan produk kontainer Anda berdasarkan negara.

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/> dan masuk ke akun penjual Anda.
2. Dari halaman [Produk Server](#), pilih produk kontainer yang ingin Anda modifikasi.
3. Dari daftar tarik-turun Permintaan perubahan, pilih Perbarui penawaran umum, lalu pilih Perbarui ketersediaan menurut negara.
4. Pilih salah satu opsi berikut:
 - Semua negara - Tersedia di semua negara yang didukung.
 - Semua negara dengan pengecualian — Tersedia di semua negara yang didukung kecuali di negara tertentu.
 - Hanya negara yang diizinkan - Tersedia bagi pembeli untuk membeli hanya di negara yang Anda tentukan.
5. Pilih Kirim untuk mengirimkan permintaan Anda untuk ditinjau.
6. Verifikasi bahwa tab Permintaan menampilkan status Permintaan sebagai Dalam peninjauan. Ketika permintaan selesai, status akan diperbarui ke Berhasil atau Gagal.

Memperbarui perjanjian lisensi pengguna akhir (EULA) Anda untuk produk kontainer AWS Marketplace

Sebagai AWS Marketplace penjual, Anda dapat memperbarui perjanjian lisensi pengguna akhir (EULA) untuk menggunakan EULA [Kontrak Standar untuk AWS Marketplace](#) atau kustom. Pembaruan yang dilakukan pada EULA berlaku untuk pengguna baru yang berlangganan pembaruan produk dan produk Anda. Untuk memperbarui EULA Anda, gunakan prosedur berikut.

1. Buka Portal Manajemen AWS Marketplace at <https://aws.amazon.com/marketplace/management/tour/> dan masuk ke akun penjual Anda.
2. Dari halaman [Produk Server](#), pilih produk kontainer yang ingin Anda modifikasi.
3. Dari daftar tarik-turun Permintaan perubahan, pilih Perbarui penawaran umum, lalu pilih Perbarui EULA.
4. Pilih Kontrak Standar untuk AWS Marketplace atau kirimkan EULA kustom Anda. Untuk EULA khusus, Anda harus memberikan kontrak dari bucket Amazon S3.
5. Pilih Kirim untuk mengirimkan permintaan Anda untuk ditinjau.
6. Verifikasi bahwa tab Permintaan menampilkan status Permintaan sebagai Dalam peninjauan. Ketika permintaan selesai, status akan diperbarui ke Berhasil atau Gagal.

Menguji dan melepaskan produk kontainer Anda AWS Marketplace

Setelah Anda menambahkan versi baru produk kontainer Anda AWS Marketplace, Anda dapat menguji produk Anda dan kemudian merilisnya ke publik. Topik ini menguraikan langkah-langkah dan proses spesifik yang diperlukan untuk menguji dan merilis produk Anda ke publik.

Topik

- [Gambar kontainer dan Helm opsi pengiriman bagan](#)
- [Opsi pengiriman EKS add-on Amazon](#)

Gambar kontainer dan Helm opsi pengiriman bagan

Bagian ini memberikan panduan tentang merilis gambar Container Anda dan Helm grafik.

Permintaan Anda untuk versi baru dibuat dan akan selesai dalam beberapa menit. Anda dapat melacak permintaan dari tab Permintaan dari halaman Produk server. Jika Anda menerima

kesalahan saat menguji atau merilis add-on, lihat tabel Kesalahan Aynchronous di [Tambahkan versi baru di Referensi](#).AWS Marketplace Catalog API

Note

Jika produk Anda saat ini diatur ke ketersediaan terbatas, hanya pembeli yang menyediakan produk yang dapat mengakses versi produk. Jika produk Anda saat ini disetel ke ketersediaan publik, semua AWS Marketplace pembeli dapat mengakses versi produk.

Jika ini adalah versi pertama Anda, produk Anda sekarang siap untuk dipublikasikan.

Opsi pengiriman EKS add-on Amazon

Bagian ini memberikan panduan tentang pengujian dan pelepasan EKS add-on Amazon Anda.

Uji add-on Anda

- Setelah Anda mengirimkan add-on, AWS Marketplace memproses permintaan Anda dan menerbitkan add-on Anda dalam keadaan terbatas untuk Anda validasi di katalog add-on Amazon. EKS Anda dapat melacak permintaan dari tab Permintaan halaman produk Server di halaman Portal Manajemen AWS Marketplace. Waktu konsumsi akan bervariasi dari 5-10 hari kerja tergantung pada volume permintaan yang kami tangani.

Saat permintaan Anda dalam status Under review, add-on masih dipublikasikan oleh AWS tim dari katalog AWS Marketplace EKS add-on Amazon. Minta perubahan status ke Sukses setelah add-on dipublikasikan ke status Terbatas. Anda dapat memulai pengujian add-on Anda setelah ini.

- Setelah add-on Anda tersedia, Anda dapat menemukannya di Wilayah Asia Pasifik (Seoul) untuk tujuan pengujian. AWS Marketplace bergantung pada keahlian Anda untuk memverifikasi fungsionalitas perangkat lunak Anda. Untuk menguji add-on Anda, Anda harus membuat EKS kluster Amazon di Wilayah Asia Pasifik (Seoul) di akun penjual tempat add-on Anda diizinkan terdaftar. Untuk menguji add-on Anda, ikuti [petunjuk terperinci ini](#). Pastikan untuk menguji pada masing-masing Kubernetes versi yang didukung perangkat lunak Anda.
- Jika Anda menawarkan produk berbayar, buat penawaran pribadi ke internal berikut Akun AWS. Akun ini membantu mengintegrasikan perangkat lunak Anda ke EKS konsol Amazon di semua iklan Wilayah AWS.

288092140294, 288092140294, 408202761791

- Jaga agar kluster pengujian Anda tetap aktif hingga AWS Marketplace menyetujui dan memindahkan versi add-on Anda ke publik.

Note

AWS Marketplace tidak akan menanggung biaya AWS infrastruktur yang dikeluarkan selama pengujian produk kontainer Anda di cluster Amazon EKS Anda. Anda dapat mengikuti mekanisme ukuran yang tepat untuk menurunkan node ke biaya pengoperasian minimal saat kami memverifikasi hasil pengujian.

Lepaskan add-on Anda ke publik

Setelah Anda memvalidasi perangkat lunak Anda melalui EKS kluster Amazon sebagai add-on, Anda dapat mengirimkan permintaan untuk merilis versi EKS add-on Amazon Anda ke publik menggunakan atau [Portal Manajemen AWS Marketplace](#) AWS Marketplace Catalog API

Untuk informasi selengkapnya, lihat [Memperbarui visibilitas untuk EKS add-on Amazon](#) di Referensi.AWS Marketplace Catalog API

Anda dapat melacak permintaan dari tab Permintaan halaman produk Server di halaman Portal Manajemen AWS Marketplace. Waktu konsumsi akan bervariasi.

Persyaratan produk berbasis kontainer untuk AWS Marketplace

AWS Marketplace mempertahankan persyaratan berikut untuk semua produk dan penawaran berbasis kontainer di. AWS Marketplace Persyaratan ini membantu mempromosikan katalog yang aman, terjaga, dan dapat dipercaya untuk pelanggan kami. Kami juga mendorong penjual untuk meninjau implementasi kontrol dan protokol tambahan yang berlaku untuk memenuhi kebutuhan produk spesifik mereka.

Semua produk dan metadata terkait ditinjau ketika dikirimkan untuk memastikan mereka memenuhi atau melampaui kebijakan saat ini AWS Marketplace . Kebijakan ini diperbarui secara berkala agar selaras dengan pedoman keamanan yang berkembang. AWS Marketplace terus memindai produk untuk memverifikasi bahwa daftar yang ada terus memenuhi setiap perubahan pada persyaratan ini. Jika suatu produk tidak sesuai, AWS Marketplace akan menghubungi penjual untuk memperbarui produk mereka untuk memenuhi standar baru. Dalam beberapa kasus, produk mungkin dibuat sementara tidak tersedia untuk pelanggan baru sampai masalah diselesaikan. Proses ini membantu menjaga keamanan dan kepercayaan AWS Marketplace platform untuk semua pengguna.

Topik

- [Kebijakan Keamanan](#)
- [Persyaratan informasi pelanggan](#)
- [Persyaratan penggunaan produk](#)
- [Persyaratan arsitektur](#)
- [Petunjuk penggunaan produk kontainer](#)
- [Persyaratan untuk produk add-on Amazon EKS](#)

Kebijakan Keamanan

Semua produk berbasis kontainer harus mematuhi persyaratan keamanan berikut:

- Gambar kontainer tidak boleh berisi kerentanan, malware, atau paket perangkat lunak End-of-Life (EoL) dan sistem operasi yang diketahui.
- Kontainer tidak boleh meminta AWS kredensi untuk mengakses AWS layanan. Ketika produk Anda perlu mengakses AWS layanan, Anda harus menggunakan salah satu dari berikut ini:
 - Peran IAM untuk akun layanan, untuk beban kerja Amazon Elastic Kubernetes Service (Amazon EKS).
 - Peran IAM untuk tugas, untuk beban kerja Amazon Elastic Container Service (Amazon ECS).
- Produk berbasis kontainer hanya memerlukan sedikit hak istimewa untuk dijalankan. Untuk informasi selengkapnya, lihat [Keamanan di Amazon Elastic Container Service](#) and [Security di Amazon EKS](#).
- Citra kontainer harus dikonfigurasi untuk dijalankan dengan hak non-root secara default.
- Kontainer tidak boleh berisi rahasia hardcode seperti kata sandi (bahkan hash) untuk pengguna dan layanan sistem, kunci pribadi, kredensi, dll.
- Otentikasi dalam layanan apa pun yang berjalan di dalam wadah tidak boleh menggunakan otentikasi berbasis kata sandi, meskipun kata sandi dibuat, disetel ulang, atau ditentukan oleh pengguna saat peluncuran. Kata sandi nol dan kosong juga tidak diizinkan.
- Gambar kontainer tidak boleh menyertakan lapisan dengan arsitektur yang tidak didukung (misalnya, metadata Kerangka Pengesahan in-toto).

Persyaratan informasi pelanggan

Semua produk berbasis kontainer harus mematuhi persyaratan informasi pelanggan berikut:

- Perangkat lunak tidak boleh mengumpulkan atau mengekspor data pelanggan tanpa sepengetahuan pelanggan dan persetujuan tegas kecuali sebagaimana disyaratkan oleh BYOL (Bring Your Own License). Aplikasi yang mengumpulkan atau mengekspor data pelanggan harus mengikuti pedoman ini:
 - Pengumpulan data pelanggan harus swalayan, otomatis, dan aman. Pembeli tidak perlu menunggu penjual menyetujui untuk menyebarkan perangkat lunak.
 - Pengumpulan data pelanggan harus konsisten dengan perjanjian Anda dengan AWS, termasuk namun tidak terbatas pada, [Syarat dan Ketentuan AWS Marketplace](#), [Ketentuan AWS Layanan](#), [Pemberitahuan AWS Privasi](#), dan [Perjanjian AWS Pelanggan](#).
 - Informasi pembayaran tidak boleh dikumpulkan.

Persyaratan penggunaan produk

Semua produk berbasis kontainer harus mematuhi persyaratan penggunaan produk berikut:

- Penjual hanya bisa mencantumkan produk yang berfungsi penuh. Produk beta atau prarilis untuk tujuan uji coba atau evaluasi tidak diperbolehkan. Pengembang, komunitas, dan edisi BYOL perangkat lunak komersial didukung jika penjual menyediakan versi berbayar yang setara AWS Marketplace dalam waktu 90 hari setelah menyediakan edisi gratis.
- Semua petunjuk penggunaan produk berbasis kontainer harus meliputi semua langkah untuk mendeploy produk berbasis kontainer. Petunjuk penggunaan harus menyediakan perintah dan sumber daya deployment yang menunjuk ke citra kontainer yang sesuai pada AWS Marketplace.
- Produk berbasis kontainer harus meliputi semua citra kontainer yang dibutuhkan pelanggan untuk menggunakan perangkat lunak. Selain itu, produk berbasis kontainer tidak boleh mengharuskan pengguna untuk meluncurkan produk menggunakan gambar apa pun dari luar AWS Marketplace (misalnya, gambar kontainer dari repositori pihak ketiga).
- Kontainer dan perangkat lunaknya harus dapat digunakan dengan cara swalayan dan tidak boleh memerlukan metode atau biaya pembayaran tambahan. Aplikasi yang memerlukan dependensi eksternal pada penerapan harus mengikuti pedoman ini:
 - Persyaratan harus diungkapkan dalam deskripsi atau petunjuk penggunaan daftar. Misalnya, Produk ini memerlukan koneksi internet untuk digunakan dengan benar. Paket-paket berikut diunduh saat penerapan: `<list of package>`
 - Penjual bertanggung jawab atas penggunaan dan memastikan ketersediaan dan keamanan semua dependensi eksternal.
 - Jika dependensi eksternal tidak lagi tersedia, produk harus dihapus AWS Marketplace juga.

- Dependensi eksternal tidak boleh memerlukan metode atau biaya pembayaran tambahan.
- Kontainer yang memerlukan koneksi berkelanjutan ke sumber daya eksternal yang tidak berada di bawah kendali langsung pembeli — misalnya, eksternal APIs atau Layanan AWS dikelola oleh penjual atau pihak ketiga — harus mengikuti pedoman ini:
 - Persyaratan harus diungkapkan dalam deskripsi atau petunjuk penggunaan daftar. Misalnya, Produk ini membutuhkan koneksi internet yang berkelanjutan. Layanan eksternal yang sedang berlangsung berikut ini diperlukan untuk berfungsi dengan baik: <list of resources>
- Penjual bertanggung jawab atas penggunaan dan memastikan ketersediaan dan keamanan semua sumber daya eksternal.
- Jika sumber daya eksternal tidak lagi tersedia, produk harus dihapus AWS Marketplace juga.
- Sumber daya eksternal tidak boleh memerlukan metode atau biaya pembayaran tambahan dan pengaturan koneksi harus otomatis.
- Perangkat lunak produk dan metadata tidak boleh berisi bahasa yang mengarahkan pengguna ke platform cloud lain, produk tambahan, atau layanan upsell yang tidak tersedia di AWS Marketplace.
- Jika produk Anda merupakan add-on untuk produk lain atau produk ISV lain, deskripsi produk Anda harus menunjukkan bahwa itu memperluas fungsionalitas produk lain dan bahwa tanpa itu, produk Anda memiliki utilitas yang sangat terbatas. Misalnya, Produk ini memperluas fungsionalitas dan tanpa itu, produk ini memiliki utilitas yang sangat terbatas<product name>. Harap dicatat bahwa mungkin memerlukan lisensi sendiri untuk fungsionalitas penuh dengan daftar ini. <product name>

Persyaratan arsitektur

Semua produk berbasis kontainer harus mematuhi persyaratan arsitektur berikut:

- Gambar kontainer sumber untuk AWS Marketplace harus didorong ke repositori Amazon Elastic Container Registry (Amazon ECR) yang dimiliki oleh AWS Marketplace Anda dapat membuat repositori ini di produk server Portal Manajemen AWS Marketplace bawah untuk setiap daftar produk kontainer Anda.
- Citra kontainer harus didasarkan pada Linux.
- [Produk berbasis kontainer berbayar harus dapat digunakan di Amazon ECS, Amazon EKS, atau AWS Fargate](#)
- Produk berbasis kontainer berbayar dengan harga kontrak dan integrasi dengan AWS License Manager harus diterapkan di Amazon EKS, Amazon ECS, Amazon EKS AWS Fargate Anywhere,

Amazon ECS Anywhere, OpenShift Red Hat Service AWS on (ROSA), cluster Kubernetes yang dikelola sendiri di lokasi, atau di Amazon Elastic Compute Cloud.

Petunjuk penggunaan produk kontainer

Saat membuat petunjuk penggunaan untuk produk kontainer Anda, ikuti langkah-langkah dan panduan di [the section called “Petunjuk penggunaan produk AMI dan kontainer”](#).

Persyaratan untuk produk add-on Amazon EKS

Add-on Amazon EKS adalah perangkat lunak yang menyediakan kemampuan operasional untuk Kubernetes aplikasi tetapi tidak spesifik untuk aplikasi. Misalnya, add-on Amazon EKS menyertakan agen observabilitas atau Kubernetes driver yang memungkinkan klaster berinteraksi dengan AWS sumber daya dasar untuk jaringan, komputasi, dan penyimpanan.

Sebagai penjual produk kontainer, Anda dapat memilih di antara beberapa opsi penerapan termasuk Amazon EKS. Anda dapat mempublikasikan versi produk Anda sebagai AWS Marketplace add-on ke dalam katalog add-on Amazon EKS. Add-on Anda muncul di konsol Amazon EKS di samping add-on yang dikelola oleh AWS dan vendor lain. Pembeli Anda dapat menggunakan perangkat lunak Anda sebagai add-on semudah mereka melakukan add-on lainnya.

Untuk informasi selengkapnya, lihat [add-on Amazon EKS](#) di Panduan Pengguna Amazon EKS.

Mempersiapkan produk kontainer Anda sebagai AWS Marketplace add-on

Untuk mempublikasikan produk kontainer Anda sebagai AWS Marketplace add-on, produk tersebut harus memenuhi persyaratan berikut:

- Produk kontainer Anda harus dipublikasikan di AWS Marketplace.
- Produk kontainer Anda harus dibangun kompatibel untuk keduanya AMD64 dan ARM64 arsitektur.
- Produk kontainer Anda tidak boleh menggunakan [model harga](#) Bring Your Own License (BYOL).

Note

BYOL tidak didukung untuk pengiriman add-on Amazon EKS.

- Anda harus mematuhi semua [persyaratan produk berbasis kontainer](#) termasuk memasukkan semua gambar dan Helm bagan kontainer ke dalam repositori AWS Marketplace Amazon ECR yang dikelola. Persyaratan ini mencakup gambar sumber terbuka, misalnya, `nginx`. Gambar dan

bagian tidak dapat di-host di repositori eksternal lainnya termasuk, namun tidak terbatas pada, [Galeri Publik Amazon ECR](#), Docker Hub dan. Quay

- Helmgrafik - Mempersiapkan dan mengemas perangkat lunak Anda sebagai Helm bagan. Kerangka add-on Amazon EKS mengubah Helm bagan menjadi manifes Kubernetes. Beberapa Helm fitur tidak didukung dalam sistem Amazon EKS. Daftar berikut menjelaskan persyaratan yang harus dipenuhi sebelum melakukan onboarding perangkat lunak Anda sebagai add-on Amazon EKS. Dalam daftar ini, semua Helm perintah menggunakan Helm versi 3.8.1:
 - Semua Capabilities objek didukung, dengan pengecualian untuk .APIVersions. .APIVersion tidak didukung untuk non-built-in kustom Kubernetes APIs.
 - Hanya Release.Namespace objek Release.Name dan yang didukung.
 - Helm kait dan lookup fungsinya tidak didukung.
 - Semua grafik dependen harus ditempatkan di dalam Helm bagan utama (ditentukan dengan file jalur repositori://...).
 - HelmBagan harus berhasil melewati Helm Lint dan Helm Template tanpa kesalahan. Perintahnya adalah sebagai berikut:
 - HelmSerat — `helm lint helm-chart`

Masalah umum termasuk bagan yang tidak dideklarasikan dalam metadata bagan induk.

Sebagai contoh, chart metadata is missing these dependencies: chart-base
Error: 1 chart(s) linted, 1 chart(s) failed.

- HelmTemplate - `helm template chart-name chart-location --set k8version=Kubernetes-version --kube-version Kubernetes-version --namespace addon-namespace --include-crds --no-hooks --f any-overridden-values`

Lewati konfigurasi yang diganti dengan bendera. --f

- Simpan semua binari kontainer di repo AWS Marketplace Amazon ECR. Untuk membuat manifes, gunakan perintah Helm template yang ditampilkan sebelumnya. Cari manifes untuk referensi gambar eksternal seperti busybox atau gcr gambar. Unggah semua gambar kontainer bersama dengan dependensi ke repo AWS Marketplace Amazon ECR yang dibuat dengan menggunakan opsi Add Repository di dropdown permintaan.
- Konfigurasi kustom - Anda dapat menambahkan variabel kustom selama penerapan. Untuk informasi tentang cara mengidentifikasi pengalaman pengguna akhir, beri nama perangkat lunak `aws_mp_configuration_schema.json`, dan paket ke dalam pembungkus dengan Helm bagan, lihat [Amazon EKS add-ons: Advanced configuration](#).

Menurut [Kata Kunci “\\$schema”](#), \$schema harus berupa URI yang menunjuk ke sumber daya yang valid `application/schema+json`.

File ini tidak boleh menerima informasi sensitif seperti kata sandi, kunci lisensi, dan sertifikat.

Untuk menangani rahasia dan instalasi sertifikat, Anda dapat memberikan langkah pasca atau pre-Add-on instalasi kepada pengguna akhir. Produk tidak boleh bergantung pada lisensi eksternal apa pun. Produk harus bekerja berdasarkan AWS Marketplace hak.

Untuk informasi selengkapnya tentang batasan `aws_mp_configuration_schema.json`, lihat [Persyaratan konfigurasi add-on dan praktik terbaik untuk penyedia add-on](#).

- Identifikasi dan buat namespace tempat perangkat lunak akan digunakan - Dalam rilis pertama produk Anda, Anda harus mengidentifikasi namespace tempat perangkat lunak akan digunakan dengan menambahkan namespace templat.
- Definisi sumber daya khusus (CRDs) - Kerangka add-on Amazon EKS tidak mendukung penginstalan CRDs dan deklarasi sumber daya khusus berdasarkan CRDs penerapan dengan add-on yang sama. Jika add-on Anda memiliki sumber daya khusus dan bergantung pada CRDs, Anda dapat:
 - Publikasikan dua add-on: Pisahkan definisi CRD menjadi add-on terpisah (bagan helm terpisah) dan instalasi [sumber daya kustom](#) yang sebenarnya menjadi add-on terpisah.
 - Publikasikan satu add-on dengan instruksi manual tambahan: Publikasikan satu add-on yang menginstal cluster on. CRDs Berikan instruksi penggunaan bersama dengan file manifes kubernetes bagi pengguna akhir untuk menyiapkan sumber daya kustom yang bergantung padanya. CRDs
- Buat **serviceAccount** jika berlaku — Jika perangkat lunak adalah perangkat lunak berbayar AWS Marketplace atau harus terhubung dengan yang lain Layanan AWS, pastikan Helm bagan dibuat secara `serviceAccount` default. Jika `serviceAccount` kreasi ditangani oleh parameter dalam `values.yaml` file, atur nilai parameter ke `true`. Misalnya, `serviceAccount.create = true`. Ini diperlukan karena pelanggan mungkin memilih untuk menginstal add-on dengan mewarisi izin dari instance node yang mendasarinya yang sudah memiliki izin yang diperlukan. Jika bagan Helm tidak membuat `serviceAccount`, maka izin tidak dapat dikaitkan dengan `serviceAccount`
- Penerapan atau Daemonset yang Dapat Dilacak — Pastikan bagan Helm Anda memiliki daemonset atau penerapan. Kerangka add-on Amazon EKS melacak penerapan sumber daya Amazon EKS Anda yang menggunakannya. Tanpa penerapan atau daemonset yang dapat dilacak, add-on Anda akan menghadapi kesalahan penerapan. Jika add-on Anda tidak memiliki

deployment atau daemonset, misalnya, jika addon Anda menerapkan sekumpulan sumber daya Kustom atau pekerjaan Kubernetes yang tidak dapat dilacak, tambahkan deployment dummy atau objek daemonset.

- Support untuk arsitektur AMD dan ARM — Banyak pelanggan Amazon EKS menggunakan ARM64 saat ini untuk menggunakan instans AWS Graviton. Perangkat lunak pihak ketiga harus mendukung kedua arsitektur.
- Integrasikan dengan lisensi atau pengukuran APIs dari AWS Marketplace - AWS Marketplace mendukung beberapa model penagihan. Untuk informasi selengkapnya, lihat [Integrasi penagihan, pengukuran, dan lisensi produk kontainer](#). Jika Anda ingin menjual produk Anda melalui mekanisme PAYG, lihat [Mengkonfigurasi pengukuran khusus untuk produk kontainer dengan AWS Marketplace Metering Service](#). Jika Anda ingin menjual produk Anda melalui model dimuka atau kontrak, lihat [Harga kontrak untuk produk kontainer dengan AWS License Manager](#).
- Unggah perangkat lunak dan semua artefak dan dependensi — Bagan Helm harus mandiri, dan tidak boleh memerlukan dependensi dari sumber eksternal, misalnya, GitHub. Jika perangkat lunak memerlukan dependensi eksternal, maka dependensi harus didorong ke repositori AWS Marketplace Amazon ECR pribadi di bawah daftar yang sama. AWS Marketplace
- Berikan instruksi penerapan di situs web Anda — [Kami meminta Anda meng-host panduan penyebaran bagi pelanggan untuk mengidentifikasi cara menerapkan perangkat lunak Anda melalui perintah create-addon](#).
- Izin tambah/peran IAM — Jika add-on Anda yang diterbitkan dari AWS Marketplace memerlukan akses ke AWS layanan, perangkat lunak Anda harus memiliki akun layanan Kubernetes yang dianotasi dengan kebijakan IAM untuk mengakses layanan. AWS Anda dapat memilih dari dua opsi untuk akun layanan Anda untuk membuat permintaan API ke AWS layanan:
 - Credentials via IRSA: Opsi ini memungkinkan perangkat lunak Anda untuk memperoleh kredensi asumsi dari Identity and Access Management (IAM) and Access Management (IAM) Role Service (IRSA). Untuk informasi selengkapnya, lihat [peran IAM untuk akun layanan](#).
 - Identitas pod Amazon EKS: Opsi ini memungkinkan perangkat lunak Anda menggunakan Identitas Pod dari pod Amazon EKS untuk membuat permintaan API ke AWS layanan. Untuk informasi selengkapnya, lihat [Pelajari cara EKS Pod Identity memberikan akses Pod ke layanan AWS](#)

Add-on Anda harus memiliki file konfigurasi tambahan bernama `aws_mp_addon_parameters.json` di tingkat atas bagan Helm, di direktori yang sama dengan skema konfigurasi kustom saat ini (`.aws_mp_configuration_schema.json`). Saat ini, file ini hanya menangani izin yang kompatibel dengan identitas pod. Format file adalah sebagai berikut:

```
{
  "permissions": {
    "isPodIdentityCompatible" : true,
    "permissionsList": [
      {
        "serviceAccount" : "String",
        "managedPolicies" : ["Policy Arn"],
      }
    ]
  }
}
```

Nama berkas: **aws_mp_addon_parameters.json**

 Note

aws_mp_addon_parameters.jsonFile ini mengaktifkan bagian Akses Add-on di halaman pengaturan konfigurasi Add-on di konsol Amazon EKS

Nama bidang	Tipe	Catatan	Nilai contoh
isPodIdentityKompatibel	Boolean	Hanya `true` yang didukung untuk saat ini. Bidang menunjukkan jika izin yang dijelaskan dalam daftar PermissionsList berikut sesuai dengan identitas pod	BETUL
ServiceAccount	String	Nama akun layanan yang akan digunakan add-on untuk mengakses izin	kpow

Nama bidang	Tipe	Catatan	Nilai contoh
Kebijakan yang Dikelola	Daftar <String>	Daftar kebijakan yang akan digunakan untuk akun layanan ini yang mungkin diasumsikan oleh add-on EKS	["arn:aws:iam::aws:policy/ReadOnlyAccess"]

Note

Pay-as-you-go Produk add-on (PAYG) dari tidak AWS Marketplace dapat menggunakan Amazon EKS Pod Identity dan harus menggunakan Peran IAM untuk Akun Layanan (IRSA) untuk kontrol akses.

- Pembaruan versi — Amazon EKS merilis versi Kubernetes baru beberapa minggu setelah rilis upstream. Karena versi cluster Amazon EKS baru tersedia secara umum, vendor memiliki waktu 45 hari untuk mengesahkan atau memperbarui perangkat lunak mereka agar kompatibel dengan rilis versi cluster Amazon EKS yang baru. Jika versi add-on Anda saat ini mendukung versi Kubernetes yang baru, validasi dan sertifikasi yang sama sehingga kami dapat memperbarui matriks kompatibilitas versi. Jika versi add-on baru diperlukan untuk mendukung rilis versi Kubernetes yang baru, silakan kirimkan versi baru untuk orientasi.
- Perangkat lunak mitra harus termasuk dalam salah satu jenis berikut atau menjadi perangkat lunak operasional yang akan meningkatkan Kubernetes atau Amazon EKS: Gitops | monitoring | logging | cert-management | policy-management | cost-management | autoscaling | storage | kubernetes-management | service-mesh | dll-backup | load-balancer | lokal-registry | jaringan | Keamanan | backup | ingress-controller | observability ingress-service-type
- Perangkat lunak tidak dapat berupa [Container Network Interface \(CNI\)](#).
- Perangkat lunak harus dijual melalui AWS Marketplace dan terintegrasi dengan Lisensi dan pengukuran APIs untuk produk berbayar. Produk BYOL tidak diterima.

Persyaratan konfigurasi add-on dan praktik terbaik untuk penyedia add-on

Amazon EKS memerlukan konfigurasi sebagai string [skema Helm JSON](#) dari penyedia add-on. Pengaya yang memerlukan konfigurasi yang diperlukan atau mengizinkan konfigurasi opsional harus menyertakan `aws_mp_configuration_schema.json` file dengan Bagan Helm yang

dikirimkan ke. AWS Marketplace Amazon EKS akan menggunakan skema ini untuk memvalidasi input konfigurasi dari pelanggan dan menolak panggilan API dengan nilai input yang tidak sesuai dengan skema. Konfigurasi add-on biasanya termasuk dalam dua kategori:

- Konfigurasi untuk properti Kubernetes umum seperti label, toleransi, nodeSelector, dll.
- Konfigurasi yang khusus add-on seperti kunci lisensi, pemberdayaan fitur, URLs dll.

Bagian ini difokuskan pada kategori pertama yang terkait dengan properti Kubernetes umum.

Amazon EKS merekomendasikan mengikuti praktik terbaik seputar konfigurasi add-on Amazon EKS.

- [Persyaratan skema](#)
- [Parameter umum yang diizinkan untuk konfigurasi](#)
- [Parameter umum yang tidak diizinkan untuk konfigurasi](#)

Persyaratan skema

Saat mendefinisikan skema json, pastikan Anda menggunakan versi jsonschema yang didukung oleh add-on Amazon EKS.

Daftar skema yang didukung:

- <https://json-schema.org/draft-04/schema>
- <https://json-schema.org/draft-06/schema>
- <https://json-schema.org/draft-07/schema>
- <https://json-schema.org/draft/2019-09/schema>

Menggunakan versi skema json lainnya tidak kompatibel dengan add-on Amazon EKS dan akan menyebabkan add-on tidak dapat dirilis hingga ini diperbaiki.

Contoh file skema Helm

```
{
  "$schema": "http://json-schema.org/schema#",
  "type": "object",
  "properties": {
    "podAnnotations": {
      "description": "Pod Annotations"
```

```
"type": "object"
  },
  "podLabels": {
    "description": "Pod Labels"
    "type": "string"
  },
  "resources": {
    "type": "object"
    "description": "Resources"
  },
  "logLevel": {
    "description": "Logging Level"
    "type": "string",
    "enum": [
      "info",
      "debug"
    ]
  },
  "config": {
    "description": "Custom Configuration"
    "type": "object"
  }
}
```

camelCase

Parameter konfigurasi harus menjadi CamelCase, dan akan ditolak jika tidak mengikuti format ini.

Deskripsi diperlukan

Selalu sertakan deskripsi yang bermakna untuk properti skema. Deskripsi ini akan digunakan untuk merender nama label di konsol Amazon EKS untuk setiap parameter konfigurasi.

Definisi RBAC

Penyedia add-on perlu mendefinisikan dan menyediakan izin RBAC yang diperlukan untuk berhasil menginstal add-on menggunakan prinsip hak istimewa paling sedikit. Jika izin RBAC perlu diubah untuk versi add-on yang lebih baru atau perbaikan apa pun untuk mengatasi CVE, penyedia add-on perlu memberi tahu tim Amazon EKS tentang perubahan ini. Izin yang diperlukan untuk setiap sumber daya Kubernetes harus dibatasi pada nama sumber daya objek.

```
apiGroups: ["apps"]
resources: ["daemonsets"]
```

```
resourceNames: ["ebs-csi-node"]
verbs: ["create", "delete", "get", "list", "patch", "update", "watch"]
```

Manajemen Rahasia

Bagian ini hanya berlaku untuk add-on yang membutuhkan pelanggan untuk mengonfigurasi informasi rahasia seperti kunci aplikasi, kunci API, kata sandi, dll. Saat ini, Amazon EKS APIs tidak mendukung penyampaian informasi rahasia dalam teks biasa karena implikasi keamanan. Namun, pelanggan dapat menggunakan konfigurasi untuk meneruskan nama Rahasia Kubernetes yang menyimpan kunci yang dibutuhkan oleh add-on. Pelanggan akan diminta untuk membuat objek Kubernetes Secret yang berisi kunci dengan namespace yang sama sebagai langkah prasyarat dan kemudian meneruskan nama Secret menggunakan gumpalan konfigurasi saat membuat add-on. Kami menyarankan agar penyedia add-on memberi nama properti skema sehingga pelanggan tidak secara tidak sengaja salah mengira itu sebagai kunci yang sebenarnya. Misalnya: appSecretName, connectionSecretName dll.

Singkatnya, penyedia add-on dapat memanfaatkan skema untuk memungkinkan pelanggan melewati nama rahasia tetapi bukan kunci yang benar-benar akan menyimpan rahasia itu sendiri.

Contoh nilai konfigurasi

Anda dapat menyertakan contoh konfigurasi dalam skema Anda untuk membantu pelanggan dengan konfigurasi add-on. Contoh berikut adalah dari skema AWS Distro untuk OpenTelemetry add-on.

```
"examples": [
  {
    "admissionWebhooks": {
      "namespaceSelector": {},
      "objectSelector": {}
    },
    "affinity": {},
    "collector": {
      "amp": {
        "enabled": true,
        "remoteWriteEndpoint": "https://aps-workspaces.us-west-2.amazonaws.com/workspaces/ws-xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxxx/api/v1/remote_write"
      },
      "cloudwatch": {
        "enabled": true
      },
    },
    "mode": "deployment",
```

```
"replicas": 1,
"resources": {
  "limits": {
    "cpu": "256m",
    "memory": "512Mi"
  },
  "requests": {
    "cpu": "64m",
    "memory": "128Mi"
  }
},
"serviceAccount": {
  "annotations": {},
  "create": true,
  "name": "adot-collector"
},
"xray": {
  "enabled": true
}
},
"kubeRBACProxy": {
  "enabled": true,
  "resources": {
    "limits": {
      "cpu": "500m",
      "memory": "128Mi"
    },
    "requests": {
      "cpu": "5m",
      "memory": "64Mi"
    }
  }
},
"manager": {
  "env": {},
  "resources": {
    "limits": {
      "cpu": "100m",
      "memory": "128Mi"
    },
    "requests": {
      "cpu": "100m",
      "memory": "64Mi"
    }
  }
}
```

```

    }
  },
  "nodeSelector": {},
  "replicaCount": 1,
  "tolerations": []
}
]

```

Parameter umum yang diizinkan untuk konfigurasi

Berikut ini adalah parameter yang direkomendasikan dalam file skema Helm yang dihadapi pelanggan.

Parameter	Deskripsi	Harus memiliki default?
TambahanAlllabels	Tambahkan label Kubernetes ke semua objek Kubernetes yang dikelola oleh add-on.	Tidak
Penambahananannotasi	Tambahkan anotasi Kubernetes ke semua objek Kubernetes yang dikelola oleh add-on.	Tidak
PodLabel	Tambahkan label Kubernetes ke pod yang dikelola oleh add-on.	Tidak
PodAnnotations	Tambahkan anotasi Kubernetes ke pod yang dikelola oleh add-on.	Tidak
logLevel	Tingkat log untuk komponen yang dikelola oleh add-on.	Ya
NodeSelector	Bentuk kendala pemilihan simpul yang direkomendasikan paling sederhana. Anda dapat menambahkan field nodeSelector ke spesifikasi Pod Anda dan menentukan label	Berpotensi, misalnya node Linux saja

Parameter	Deskripsi	Harus memiliki default?
	node yang Anda inginkan untuk dimiliki oleh node target.	
toleransi	Toleransi diterapkan pada polong. Toleransi memungkinkan scheduler untuk menjadwalkan pod dengan taints yang cocok. Toleransi memungkinkan penjadwalan tetapi tidak menjamin penjadwalan.	Mungkin, lebih umum dengan daemonset
afinitas	Fitur afinitas terdiri dari dua jenis afinitas: Fungsi afinitas node seperti bidang NodeSelector tetapi lebih ekspresif dan memungkinkan Anda untuk menentukan aturan lunak, afinitas/anti-afinitas antar Pod memungkinkan Anda untuk membatasi Pod terhadap label pada Pod lain.	Mungkin
topologySpreadConstraints	Anda dapat menggunakan batasan penyebaran topologi untuk mengontrol bagaimana Pod tersebar di seluruh kluster Anda di antara domain kegagalan seperti wilayah, zona, node, dan domain topologi yang ditentukan pengguna lainnya. Ini dapat membantu mencapai ketersediaan tinggi serta pemanfaatan sumber daya yang efisien.	Mungkin
permintaan/batas sumber daya	Tentukan berapa banyak cpu/memori yang dibutuhkan setiap wadah. Permintaan sangat disarankan untuk diatur. Batas bersifat opsional.	Ya

Parameter	Deskripsi	Harus memiliki default?
replika	Jumlah replika pod yang dikelola oleh add-on. Tidak berlaku untuk daemonset.	Ya

Note

Untuk parameter konfigurasi penjadwalan beban kerja, Anda mungkin perlu memisahkan komponen tingkat atas dalam Skema jika diperlukan. Contoh, driver Amazon EBS CSI berisi dua komponen utama, controller dan node agent - pelanggan memerlukan pemilih simpul/ toleransi yang berbeda untuk setiap komponen.

Note

Nilai default yang didefinisikan dalam skema JSON murni untuk tujuan dokumentasi pengguna saja dan tidak menggantikan kebutuhan untuk memiliki default yang sah dalam file. `values.yaml` Jika menggunakan properti default, pastikan bahwa default `values.yaml` cocok dengan skema dan dua artefak (`values.schema.json` dan `values.yaml`) tetap sinkron setiap kali perubahan dilakukan pada Bagan Helm.

```
"affinity": {
  "default": {
    "affinity": {
      "nodeAffinity": {
        "preferredDuringSchedulingIgnoredDuringExecution": [
          {
            "preference": {
              "matchExpressions": [
                {
                  "key": "eks.amazonaws.com/compute-type",
                  "operator": "NotIn",
                  "values": [
                    "fargate"
                  ]
                }
              ]
            }
          }
        ]
      }
    }
  }
}
```

```

        }
      ]
    },
    "weight": 1
  }
]
},
"podAntiAffinity": {
  "preferredDuringSchedulingIgnoredDuringExecution": [
    {
      "podAffinityTerm": {
        "labelSelector": {
          "matchExpressions": [
            {
              "key": "app",
              "operator": "In",
              "values": [
                "ebs-csi-controller"
              ]
            }
          ]
        },
        "topologyKey": "kubernetes.io/hostname"
      },
      "weight": 100
    }
  ]
}
},
"description": "Affinity of the controller pod",
"type": [
  "object",
  "null"
]
}
}

```

Parameter umum yang tidak diizinkan untuk konfigurasi

Parameter metadata cluster seperti `clusterName`, `region`, `vpcId`, `accountId`, dan lainnya mungkin diperlukan oleh berbagai add-on (misalnya, Elastic Load Balancing Controller). Parameter apa pun yang mirip dengan ini yang diketahui oleh layanan Amazon EKS akan secara otomatis

disuntikkan oleh add-on Amazon EKS, dan tidak bertanggung jawab pengguna untuk menentukan sebagai opsi konfigurasi. Parameter ini meliputi:

- AWS wilayah
- Nama cluster Amazon EKS
- ID VPC dari cluster
- Registri kontainer, khusus untuk akun build-prod, yang digunakan oleh add-on jaringan
- IP cluster DNS, khusus untuk add-on coredns
- Titik akhir API kluster Amazon EKS
- IPv4 diaktifkan pada cluster
- IPv6 diaktifkan pada cluster
- Delegasi awalan untuk IPv6 diaktifkan di cluster

Penyedia add-on perlu memastikan Anda memiliki template yang ditentukan untuk parameter yang berlaku tersebut. Masing-masing parameter di atas akan memiliki `parameterType` atribut yang telah ditentukan sebelumnya yang ditentukan oleh Amazon EKS. Metadata rilis akan menentukan pemetaan antara `parameterType` name/path of the parameter in the template. This way, the values can be dynamically passed-in by Amazon EKS without requiring customers to specify these through configurations and also gives flexibility to add-on providers to define their own template name/path Parameter seperti di atas yang perlu disuntikkan Amazon EKS secara dinamis harus dikecualikan dari file skema.

Contoh pemetaan dari metadata rilis

```
"defaultConfiguration": [  
  {  
    "key": "image.containerRegistry",  
    "parameterType": "CONTAINER_REGISTRY"  
  }  
]
```

Berikut ini adalah parameter yang tidak disarankan untuk dapat dikonfigurasi dalam file skema Helm yang dihadapi pelanggan. Entah parameter harus memiliki default yang tidak dapat dimodifikasi, atau tidak disertakan sama sekali dalam template add-on.

Parameter	Deskripsi	Harus memiliki default?
gambar	Container image yang akan di-deploy pada cluster Kubernetes.	Tidak, dikelola melalui definisi add-on
imagePullSecrets	Mengkonfigurasi pod untuk menggunakan rahasia untuk menarik dari registri pribadi.	N/A
LivenessProbe	Proses Kubelet menggunakan liveness probe untuk mengetahui kapan harus me-restart sebuah container. Misalnya, liveness probe dapat menangkap kebuntuan, di mana aplikasi berjalan, tetapi tidak dapat membuat kemajuan. Memulai ulang wadah dalam keadaan seperti itu dapat membantu membuat aplikasi lebih tersedia meskipun ada bug.	Ya
ReadinessProbe	Penting bahwa Anda memiliki probe kesiapan untuk wadah Anda. Dengan cara ini proses Kubelet yang berjalan di pesawat data Anda akan tahu kapan container siap melayani lalu lintas. Sebuah Pod dianggap siap ketika semua kontainernya sudah siap. Salah satu penggunaan sinyal ini adalah untuk mengontrol Pod mana yang digunakan sebagai backend untuk Layanan. Ketika sebuah Pod tidak siap, Pod akan dihapus dari Service load balancer.	Ya

Parameter	Deskripsi	Harus memiliki default?
StartupProbe	<p>Kubelet menggunakan probe startup untuk mengetahui kapan aplikasi container telah dimulai. Jika probe seperti itu dikonfigurasi, itu menonaktifkan pemeriksaan keaktifan dan kesiapan hingga berhasil, memastikan probe tersebut tidak mengganggu startup aplikasi. Ini dapat digunakan untuk mengadopsi pemeriksaan keaktifan pada kontainer awal yang lambat, menghindari mereka dibunuh oleh kubelet sebelum mereka aktif dan berjalan.</p>	Opsional
podDisruptionBudget	<p>Tentukan Pod Disruption Budget (PDB) untuk memastikan jumlah minimum POD tetap berjalan selama gangguan sukarela. PDB membatasi jumlah Pod dari aplikasi yang direplikasi yang turun secara bersamaan dari gangguan sukarela. Misalnya, aplikasi berbasis kuorum ingin memastikan bahwa jumlah replika yang berjalan tidak pernah dibawa di bawah angka yang dibutuhkan untuk kuorum. Front end web mungkin ingin memastikan bahwa jumlah replika yang melayani beban tidak pernah turun di bawah persentase tertentu dari total.</p>	Ya, jika default ke lebih dari dua replika
ServiceAccount (nama)	Nama pod akun layanan akan berjalan di bawah.	Ya

Parameter	Deskripsi	Harus memiliki default?
ServiceAccount (anotasi)	Anotasi diterapkan ke akun layanan. Biasanya digunakan untuk fitur IAM Roles for Service Accounts	Tidak, peran akun layanan IAM ARN diatur di API add-on Amazon EKS tingkat atas. Pengecualian untuk aturan ini adalah jika add-on Anda memiliki beberapa penerapan /pengontrol (seperti Flux) dan memerlukan peran IRSA terpisah. ARNs
priorityClassName	Prioritas menunjukkan pentingnya sebuah Pod relatif terhadap Pod lainnya. Jika Pod tidak dapat dijadwalkan, scheduler akan mencoba untuk mencegah (mengusir) Pod prioritas yang lebih rendah untuk memungkinkan penjadwalan Pod yang tertunda.	Ya. Sebagian besar add-on sangat penting untuk fungsionalitas cluster, dan harus memiliki kelas prioritas yang ditetapkan secara default.

Parameter	Deskripsi	Harus memiliki default?
podSecurityContext	Konteks keamanan mendefinisikan pengaturan hak istimewa dan kontrol akses untuk Pod atau Container. Biasanya digunakan untuk mengatur FSGroup - yang diperlukan untuk IRSA di v1.19 dan cluster yang lebih rendah.	Tidak mungkin, mengingat Amazon EKS tidak lagi mendukung Kubernetes v1.19
SecurityContext	Konteks keamanan mendefinisikan pengaturan hak istimewa dan kontrol akses untuk Pod atau Container.	Ya
UpdateStrategy	Menentukan strategi yang digunakan untuk mengganti Pod lama dengan yang baru.	Ya
NameOverride	Ganti nama pod.	Tidak
podSecurityPolicy	Menegakkan pembatasan pada parameter.	Tidak - tidak PSPs digunakan lagi
extraVolumeMounts/Ekstravolum	Digunakan untuk IRSA di kluster EKS non Amazon.	Tidak

Harga produk kontainer untuk AWS Marketplace

Pada AWS Marketplace, Anda dapat membuat daftar produk gratis, produk Bring Your Own License model (BYOL), dan produk berbayar untuk Amazon Elastic Container Service (Amazon ECS) Service (Amazon ECS), Amazon Elastic Kubernetes Service (Amazon EKS), dan AWS Fargate Anda hanya dapat menetapkan satu harga per produk. Topik ini menguraikan model harga yang tersedia untuk produk kontainer.

Note

Anda menggunakan [Layanan AWS Marketplace Pengukuran](#) untuk menegakkan hak dan penggunaan meter untuk produk berbayar Anda. Untuk per tugas atau harga per pod, penggunaan diukur secara otomatis oleh AWS.

Harga yang Anda tetapkan untuk produk kontainer berlaku untuk semua Wilayah AWS. Setiap kali Anda menurunkan harga untuk produk kontainer, harga baru diterapkan untuk pembeli Anda segera. Untuk kenaikan harga, pembeli yang ada diberitahu tentang perubahan 90 hari sebelum berdampak pada penagihan mereka. Pembeli baru ditagih jumlah baru.

Note

Untuk pelanggan baru, perubahan harga segera efektif. Untuk pelanggan yang sudah ada, perubahan harga berlaku efektif pada hari pertama bulan setelah periode 90 hari yang dimulai pada tanggal pemberitahuan perubahan harga dikirim. Misalnya, Anda mengirim pemberitahuan perubahan harga pada 16 Maret. 16 Juni sekitar 90 hari setelah 16 Maret. Karena perubahan harga terjadi pada hari pertama bulan yang mengikuti periode 90 hari, tanggal efektif perubahan adalah 1 Juli.

Topik

- [Model harga kontainer](#)
- [Harga kontrak untuk produk kontainer](#)

Model harga kontainer

AWS Marketplace memiliki beberapa model harga untuk produk kontainer.

Tabel berikut memberikan informasi umum tentang model penetapan harga untuk produk berbasis kontainer.

Model harga untuk produk kontainer

Model penentuan harga	Deskripsi
Bawa Lisensi Sendiri (BYOL)	BYOL dikelola di luar AWS Marketplace melalui hubungan penagihan eksternal yang Anda pertahankan dengan pembeli.
Bulanan	<p>Harga bulanan tetap</p> <p>Harga bulanan tetap yang memberi pengguna penggunaan produk tanpa batas selama bulan berikutnya.</p> <p>Contoh: Anda menetapkan harga untuk produk Anda pada \$99 per bulan. Produk Anda menyertakan tiga gambar kontainer berbeda yang digunakan menggunakan definisi tugas Amazon ECS.</p> <p>Setelah pembeli berlangganan produk Anda, mereka segera dikenakan biaya \$99, yang berulang setiap bulan sampai mereka membatalkan langganan. Pembeli juga mendapatkan penggunaan produk yang tidak terbatas. Pembeli juga membayar secara terpisah untuk infrastruktur apa pun yang menjalankan tugas. Saat berlangganan, mereka dapat mengakses gambar kontainer Anda. Mereka dapat meluncurkan dan menjalankan sejumlah kontainer dari gambar-gambar tersebut di Amazon ECS atau Amazon EKS dalam konfigurasi apa pun.</p> <p>Jika pembeli membatalkan langganan mereka di pertengahan bulan, mereka kehilangan akses ke repositori Amazon ECR tempat AWS Marketplace menyimpan gambar kontainer. Pembeli mungkin telah menarik dan menyimpan gambar asli. Namun, mereka tidak dapat lagi menarik versi gambar kontainer baru yang Anda sediakan AWS Marketplace. Pembeli dikembalikan untuk bagian yang tidak digunakan pada bulan terakhir. Anda dibayar berdasarkan penggunaan pembeli dikurangi biaya yang disepakati AWS Marketplace .</p>
Dimensi harga metrik kustom	<p>Harga terukur khusus berdasarkan dimensi yang Anda tentukan (misalnya pengguna, node, repositori, atau GB), hingga 24 dimensi per produk.</p> <p>Contoh: Biaya produk Anda oleh pengguna. Anda memiliki pengguna admin dan pengguna reguler, dan Anda menentukan harga sebagai \$2 untuk</p>

Model penentuan harga	Deskripsi
	<p>pengguna admin dan \$1 untuk pengguna reguler. Anda dapat mengaturnya sebagai dimensi terpisah saat mencantumkan produk Anda. Anda mengenakan biaya oleh pengguna yang masuk per hari dan Anda mengukur penggunaan itu per hari.</p>
<p>Per tugas atau per pod harga per jam</p>	<p>Tugas Amazon ECS atau pod Amazon EKS</p> <p>Per tugas Amazon ECS atau per Amazon EKS harga pod yang kami ukur hingga yang kedua dengan harga yang ditetapkan per jam.</p> <p>Contoh: Produk Anda mencakup tiga gambar kontainer yang berbeda: simpul pengontrol, node pekerja, dan simpul analitik. Karena produk Anda tidak berfungsi atau berguna tanpa node pengontrol, Anda memutuskan bahwa itu adalah gambar yang ingin Anda isi untuk penggunaan. Anda menetapkan harga \$6 per jam.</p> <p>Anda memodifikasi perangkat lunak dalam image kontainer agar node pengontrol dapat diintegrasikan dengan operasi AWS Marketplace Metering Service RegisterUsage API. Ini memastikan bahwa hanya pembeli dengan langganan aktif yang dapat meluncurkan dan menjalankan gambar kontainer itu dan penggunaannya diukur berdasarkan berapa lama itu berjalan.</p> <p>Pembeli dikenakan biaya \$6 per jam penggunaan untuk setiap pod pengontrol Amazon EKS yang berjalan. Jika pembeli meluncurkan lima pod pengontrol Amazon EKS yang menyertakan kontainer node pengontrol, mereka dikenakan biaya \$30 per jam (\$6 per pod). Pembeli juga membayar secara terpisah untuk infrastruktur apa pun yang dijalankan pod.</p> <p>Untuk harga per jam, penagihan per detik dengan minimum 1 menit. Jika pelanggan menjalankan kontainer pengontrol ini selama 20 menit dan 30 detik, mereka akan terisi $20 \times (\\$6/60) + 30 \times (\\$6/60/60) = \\$2 + \\$0.05 = \\$2.05$. Anda dibayar berdasarkan penggunaan pembeli dikurangi biaya yang disepakati AWS Marketplace .</p>

Model penentuan harga	Deskripsi
<p>Setiap jam/penggunaan dengan kontrak jangka panjang</p>	<p>Kontrak jangka panjang, dengan harga yang lebih murah, dibayar di muka atau dengan angsuran reguler. Kontrak jangka panjang dapat ditambahkan ke produk yang sudah ada yang memiliki harga terukur khusus, atau per tugas dan harga per pod. Pembeli membayar harga terukur ketika mereka mengonsumsi lebih dari apa yang mereka beli dalam kontrak jangka panjang.</p> <p>Contoh: Untuk model harga terukur, Anda dapat menambahkan harga kontrak jangka panjang bagi pembeli untuk mendapatkan diskon karena melakukan di muka. Katakanlah bahwa Anda biasanya mengenakan biaya \$1 per beberapa unit yang dikonsumsi. Pembeli yang menggunakan 1 unit per jam akan membayar \$8760 per tahun ($365 \text{ days} \times 24 \text{ hours} \times \\1 per hour). Anda dapat mengaktifkan kontrak yang memungkinkan pembeli untuk menggunakan 1 unit per jam selama 365 hari dengan setengah harga itu (\$4380). Dalam hal ini, pembeli berkomitmen untuk membayar di muka untuk kontrak satu tahun, dan harga turun dari \$1 per unit menjadi \$0,5 per unit. Anda juga dapat memungkinkan pembeli untuk membeli beberapa kontrak ini. Jika kuantitas yang diukur menunjukkan bahwa pembeli mengonsumsi 10 unit dalam satu jam, dan mereka memiliki dua kontrak, maka 2 unit akan dimasukkan dalam 2 kontrak. 8 unit tambahan akan ditagih pada reguler \$1 per jam, dengan total \$8 pada jam itu.</p> <p>Untuk contoh per tugas atau per pod, Anda juga dapat menambahkan harga kontrak jangka panjang bagi pembeli untuk mendapatkan diskon untuk melakukan di muka. Jika Anda biasanya mengenakan biaya \$6 per pod, Anda dapat menetapkan durasi kontrak jangka panjang 365 hari dengan harga \$13.140 ($365 \text{ days} \times 24 \text{ hours} \times \\$3 \text{ per pod per hour}$). Satu kontrak kemudian akan memberikan hak kepada pelanggan untuk 1 pod per jam selama 365 hari tersebut. Pelanggan dapat memilih untuk membeli beberapa kontrak. Misalnya, pelanggan dapat membeli dua kontrak yang memberikan hak kepada mereka untuk 2 pod per jam. Jika pelanggan menjalankan lebih banyak pod per jam daripada kontrak yang berhak, maka kelebihan pod akan ditagih dengan harga normal per jam Anda.</p>

Model penentuan harga	Deskripsi
	Dalam kedua kasus tersebut, pembeli yang membeli kontrak jangka panjang akan ditagih di muka, baik sebagai pembayaran satu kali atau pembayaran future yang dijadwalkan secara teratur. Pembeli juga akan ditagih untuk penggunaan tambahan apa pun di atas kontrak mereka dengan tarif terukur.
Harga kontrak kontainer	Kontainer dengan harga kontrak — Produk berbasis kontainer yang pembeli membayar biaya di muka.

Harga kontrak untuk produk kontainer

Untuk produk berbasis kontainer dengan harga kontrak, AWS Marketplace menagih pelanggan Anda di muka atau dengan jadwal pembayaran yang Anda tentukan, berdasarkan kontrak antara Anda dan pelanggan Anda. Setelah itu, mereka berhak menggunakan sumber daya tersebut.

Untuk menetapkan harga Anda, pilih satu atau lebih durasi kontrak yang Anda tawarkan kepada pelanggan. Anda dapat memasukkan harga yang berbeda untuk setiap durasi kontrak. Pilihan Anda adalah durasi 1 bulan, 12 bulan, 24 bulan, dan 36 bulan. Untuk penawaran pribadi, Anda dapat menentukan durasi khusus dalam beberapa bulan (hingga 60 bulan).

Pilih kategori yang paling menggambarkan harga produk Anda. Kategori harga muncul untuk pelanggan di situs AWS Marketplace web. Anda dapat memilih dari Bandwidth (GB/s, MB/s), Data (GB, MB, TB), Host, Requests, Tiers, atau Users. Jika tidak ada kategori standar yang sesuai dengan kebutuhan Anda, Anda dapat memilih kategori Unit yang lebih umum.

Penawaran ini memungkinkan hingga 24 dimensi untuk ditambahkan ke dalamnya. Setiap dimensi membutuhkan data berikut:

- **Kategori Kontrak** — Kategori kontrak digunakan untuk mengukur atau mengukur produk Anda jika produk mendukung pengukuran berbasis konsumsi di atas harga kontrak. Untuk produk kontrak tanpa harga berbasis konsumsi, Anda dapat memilih kategori yang paling mirip dengan kategori dimensi dalam kontrak. Jika tidak ada nilai yang menyerupai unit untuk dimensi dalam kontrak, pilih `Units`.
- **Unit Kontrak** — Unit kontrak digunakan bersama dengan kategori untuk pengukuran jika produk mendukung pengukuran berbasis konsumsi. Pilih salah satu nilai yang tersedia untuk unit yang sangat cocok dengan dimensi Anda berdasarkan kategori yang dipilih.

- **Dimensi Kontrak Memungkinkan Beberapa Pembelian** — Bidang ini digunakan untuk menunjukkan apakah penawaran adalah penawaran harga berjenjang atau penawaran non-berjenjang yang memungkinkan pembelian beberapa dimensi.

Penawaran berjenjang - Memungkinkan pembeli untuk berlangganan hanya salah satu dimensi yang tersedia dalam penawaran. Dimensi dalam penawaran berjenjang tidak memiliki konsep kuantitas. Menandatangani kontrak dengan dimensi tertentu pada dasarnya menunjukkan bahwa pembeli telah memilih fitur spesifik yang ditunjukkan oleh dimensi itu.

Penawaran tidak berjenjang — Memungkinkan pelanggan untuk mendapatkan lebih dari satu dimensi sebagai bagian dari kontrak dan memungkinkan mereka untuk mendapatkan beberapa unit dari setiap dimensi tersebut.

Menetapkan nilai true untuk bidang ini menunjukkan bahwa penawaran tersebut adalah penawaran yang tidak berjenjang. Menetapkan nilai false untuk bidang ini menunjukkan bahwa penawaran tersebut adalah penawaran berjenjang.

Saat menggunakan Product Load Form (PLF) untuk membuat kontrak untuk produk Container Anda, Anda harus menentukan bidang berikut untuk dimensi harga Anda:

- **Contracts DimensionX API Name** — Nama yang akan muncul dalam lisensi yang dihasilkan di akun pembeli. AWS License Manager Nama ini juga digunakan sebagai nilai untuk Name in Entitlement dalam panggilan CheckoutLicense API.
- **Contracts DimensionX Display Name** — Nama dimensi yang menghadap pelanggan yang akan ditampilkan pada detail produk dan halaman pengadaan situs web. AWS Marketplace Buat nama yang ramah pengguna Panjang maksimum nama adalah 24 karakter. Setelah daftar bersifat publik, nilai tidak Name dapat diubah.
- **Deskripsi DimensionX Kontrak** — Deskripsi dimensi yang dihadapi pelanggan yang memberikan informasi tambahan tentang dimensi produk, seperti kemampuan yang disediakan dimensi tertentu. Panjang maksimum untuk deskripsi adalah 70 karakter.
- **Contracts DimensionX Quantity** — Ini digunakan untuk menghitung prorasi dalam kasus amandemen perjanjian terhadap suatu produk. Nilai bidang ini harus diatur ke 1 untuk semua penawaran kontrak. Seharusnya tidak diedit.
- **Kontrak DimensionX Tarif 1 Bulan** — Tarif kontrak yang akan dibebankan untuk hak 1 bulan terhadap dimensi ini. Untuk penawaran non-tier, tarif ini dibebankan untuk setiap unit dimensi yang diperoleh. Bidang ini mendukung delapan tempat desimal.

- **Kontrak DimensionX 12-Bulan Tarif** — Tarif kontrak yang akan dibebankan selama 12 bulan hak terhadap dimensi tersebut. Untuk penawaran non-tier, tarif ini dibebankan untuk setiap unit dimensi yang diperoleh. Bidang ini mendukung delapan tempat desimal.
- **Kontrak DimensionX Tarif 24 Bulan** — Tarif kontrak yang akan dibebankan selama 24 bulan hak terhadap dimensi tersebut. Untuk penawaran non-tier, tarif ini dibebankan untuk setiap unit dimensi yang diperoleh. Bidang ini mendukung delapan tempat desimal.
- **Kontrak DimensionX 36-Month Rate** — Tarif kontrak yang akan dibebankan selama 36 bulan hak terhadap dimensi tersebut. Untuk penawaran non-tier, tarif ini dibebankan untuk setiap unit dimensi yang diperoleh. Bidang ini mendukung delapan tempat desimal.

Contoh: Aplikasi penyimpanan data

	Harga 1 bulan	Harga 12 bulan	Harga 24 bulan	Harga P36 bulan
Data tidak terenkripsi (GB)	\$1,50/GB	\$16,00/GB	\$30.00/GB	\$60.00/GB
Data terenkripsi (GB)	\$1,55/GB	\$16,60/GB	\$31.20/GB	\$61.20/GB

Contoh: Produk pemantauan log

	Harga 1 bulan	Harga 12 bulan	Harga 24 bulan	Harga 36 bulan
Dasar (10 host dipantau, 5 kontainer dipantau)	\$100	\$1000	\$2000	\$4000
Standar (20 host dipantau, 10 kontainer dipantau)	\$200	\$2000	\$4000	\$8000
Pro (40 host dipantau,	\$400	\$4000	\$8000	\$16.000

	Harga 1 bulan	Harga 12 bulan	Harga 24 bulan	Harga 36 bulan
20 kontainer dipantau)				
Host tambahan dipantau per jam	\$10	\$100	\$200	\$400
Kontainer tambahan dipantau per jam	\$10	\$100	\$200	\$400

Note

Harga bisa untuk jangka waktu berikut: 1 bulan, 12 bulan, 24 bulan, atau 36 bulan. Anda dapat memilih untuk menawarkan satu atau lebih opsi ini untuk produk Anda. Durasi harus sama di setiap dimensi.

Example

Misalnya, dalam kasus di mana Anda memiliki `ReadOnlyUsers` dan `AdminUsers` dimensi, jika Anda menawarkan harga tahunan untuk `ReadOnlyUsers`, Anda harus menawarkan harga tahunan juga. `AdminUsers`

Perpanjangan otomatis

Ketika pelanggan membeli produk Anda melalui AWS Marketplace menggunakan kontrak kontainer, mereka dapat setuju untuk memperbarui persyaratan kontrak secara otomatis. Pelanggan terus membayar hak setiap bulan atau selama 1, 2, atau 3 tahun.

Pelanggan dapat mengubah pengaturan perpanjangan mereka kapan saja. Untuk informasi selengkapnya, lihat [Memodifikasi kontrak yang ada](#) di Panduan AWS Marketplace Pembeli.

Ketika kontrak kontainer berakhir

Produk kontrak kontainer memiliki kontrak kedaluwarsa. Ketika kontrak berakhir, peristiwa berikut terjadi:

1. Produk kontainer Anda menerima entitlement-updated pemberitahuan yang menunjukkan bahwa hak pembeli telah berubah, dan AWS Marketplace Entitlement Service mengembalikan respons kosong.
2. Anda memiliki satu jam untuk mengukur sisa penggunaan untuk pelanggan. Setelah ini, Anda tidak dapat lagi mengirim catatan pengukuran untuk pelanggan ini.

Integrasi penagihan, pengukuran, dan lisensi produk kontainer

AWS Marketplace terintegrasi dengan yang lain Layanan AWS untuk memberikan pengukuran dan harga berbasis kontrak untuk produk kontainer Anda. Untuk produk berbasis kontainer dengan harga penggunaan, Anda dapat menggunakan [AWS Marketplace Metering Service](#) untuk memeriksa hak untuk menggunakan produk Anda dan penggunaan pengukuran untuk penagihan. Untuk produk berbasis kontainer dengan harga kontrak, Anda dapat menggunakan lisensi AWS License Manager untuk mengaitkan lisensi dengan produk Anda. Bagian berikut memberikan informasi lebih lanjut tentang pengukuran per jam dan kustom dengan AWS Marketplace Metering Service dan harga kontrak dengan AWS License Manager

Topik

- [Pengukuran per jam dan kustom dengan AWS Marketplace Metering Service](#)
- [Harga kontrak dengan AWS License Manager](#)
- [Mengkonfigurasi pengukuran per jam dengan AWS Marketplace Metering Service](#)
- [Mengkonfigurasi pengukuran khusus untuk produk kontainer dengan AWS Marketplace Metering Service](#)
- [Harga kontrak untuk produk kontainer dengan AWS License Manager](#)

Pengukuran per jam dan kustom dengan AWS Marketplace Metering Service

[Untuk memeriksa hak untuk menggunakan produk Anda dan untuk mengukur penggunaan untuk penagihan, gunakan Layanan Pengukuran AWS Marketplace](#) . Jika Anda ingin menentukan unit harga Anda sendiri dan meteran yang digunakan kepada kami untuk penagihan, integrasikan dengan menggunakan operasi [MeterUsage](#) API. Jika Anda ingin menentukan harga produk berdasarkan jumlah tugas atau pod yang digunakan dan memiliki AWS pengukur penggunaan secara otomatis, integrasikan dengan menggunakan operasi [RegisterUsage](#) API. Untuk kedua jenis harga, Anda dapat

menambahkan harga kontrak jangka panjang tanpa mengubah cara Anda berintegrasi dengan AWS Marketplace Metering Service.

Saat Anda membuat produk kontainer baru di Portal Manajemen AWS Marketplace, kami menyediakan satu set pengidentifikasi produk (kode produk dan kunci publik) yang digunakan untuk mengintegrasikan produk Anda dengan produk. AWS Marketplace Metering Service

Penetapan

Mengintegrasikan dengan AWS Marketplace Metering Service memungkinkan Anda untuk memverifikasi bahwa pelanggan yang menjalankan perangkat lunak berbayar Anda berlangganan produk Anda AWS Marketplace, melindungi Anda dari penggunaan yang tidak sah saat startup kontainer. Untuk memverifikasi hak, gunakan operasi [MeterUsage](#) atau [RegisterUsageAPI](#), tergantung pada model harga Anda. Untuk model harga bulanan per jam dan tetap, gunakan operasi `RegisterUsage API`. Untuk model harga pengukuran kustom, gunakan operasi `MeterUsage API`.

Jika pembeli tidak berhak atas produk Anda, operasi API ini mengembalikan `CustomerNotEntitledException` pengecualian.

Note

Jika pembeli berhenti berlangganan dari produk Anda saat menjalankannya, mereka berhak untuk terus menjalankannya. Namun, mereka tidak dapat meluncurkan kontainer tambahan untuk produk Anda.

Pedoman integrasi

Saat Anda membuat dan memublikasikan produk kontainer dan menggunakan operasi `MeterUsage` atau `RegisterUsage API` untuk hak dan pengukuran, ingatlah pedoman berikut:

- Jangan mengonfigurasi AWS kredensi dalam perangkat lunak Anda atau gambar kontainer Docker. AWS kredensi untuk pembeli diperoleh secara otomatis saat runtime saat image container Anda berjalan dalam tugas Amazon ECS atau pod Amazon EKS.
- Untuk memanggil operasi `MeterUsage` atau `RegisterUsage API` dari Amazon EKS, Anda harus [menggunakan AWS SDK yang didukung](#). Untuk menguji `MeterUsage` atau `RegisterUsage` integrasi Amazon EKS, Anda harus menjalankan cluster Amazon EKS menjalankan Kubernetes 1.13.x atau lebih besar. Kubernetes 1.13 diperlukan untuk peran AWS Identity and Access Management (IAM) untuk dukungan pod. Peran IAM

diperlukan untuk pod yang sedang berjalan untuk mendapatkan AWS kredensial yang diperlukan untuk menjalankan tindakan ini di Amazon EKS.

- Anda dapat melakukan pengembangan lokal, tetapi Anda akan mendapatkan `PlatformNotSupportedException` pengecualian. Pengecualian ini tidak akan terjadi saat Anda meluncurkan penampung pada layanan AWS kontainer (Amazon ECS, Amazon EKS, dan Fargate).

Didukung Wilayah AWS

Untuk daftar semua yang AWS Marketplace didukung Wilayah AWS, lihat [Tabel Wilayah](#) di situs web Infrastruktur Global.

Memperoleh Wilayah AWS untuk metering

Saat mengintegrasikan penampung Anda untuk pengukuran dengan operasi `MeterUsage` atau `RegisterUsage` API, jangan mengonfigurasi AWS SDK untuk menggunakan yang spesifik. Wilayah AWS Wilayah harus diperoleh secara dinamis saat runtime.

Example

Misalnya, pelanggan meluncurkan tugas Amazon ECS atau pod Amazon EKS. Operasi `RegisterUsage` API dipanggil di Wilayah yang berbeda dari Wilayah tempat tugas Amazon ECS atau pod Amazon EKS diluncurkan. Oleh karena itu, operasi `RegisterUsage` API menimbulkan `InvalidRegionException` kesalahan.

AWS Bahasa SDK tidak menentukan `AWS_REGION` secara konsisten. Jika SDK Anda tidak secara otomatis mengambil `AWS_REGION`, perangkat lunak perlu ditulis secara manual untuk menentukan `AWS_Region`. Misalnya, [metadata EC2 instans Amazon AWS SDK untuk Java](#) secara otomatis (khususnya, `ec2InstanceMetadata`) untuk mendapatkan Wilayah ketika variabel lingkungan atau konfigurasi lain tidak ada. Dalam hal ini, hanya panggilan `ec2InstanceMetadata` jika `AWS_REGION` variabel lingkungan tidak ada.

Untuk informasi tentang cara mendapatkan Wilayah AWS saat runtime secara dinamis, lihat [Panduan Pengembang AWS SDK](#) untuk bahasa pemrograman Anda.

Mencegah modifikasi pengukuran

Memperkenalkan cara bagi pembeli untuk mengubah atau mengganti panggilan `keRegisterUsage` atau `MeterUsage` dapat mengakibatkan masalah penagihan dan pembayaran

yang tidak diinginkan. Kami sangat menyarankan Anda mengintegrasikan logika pengukuran dan hak.

Saat merekayasa produk Anda untuk mencegah modifikasi pengukuran, ingatlah hal berikut:

- Jika pembeli dapat menyisipkan lapisan gambar baru yang berisi CMD atau ENTRYPOINT instruksi, langsung MeterUsage integrasikan RegisterUsage atau ke dalam perangkat lunak yang dijalankan pembeli melalui gambar kontainer Anda. Jika tidak, panggilan ke RegisterUsage atau MeterUsage dieksekusi melalui CMD atau ENTRYPOINT dari gambar dasar kemungkinan akan diganti oleh pembeli.
- Kami menyarankan Anda mengelola kode AWS Marketplace produk yang digunakan perangkat lunak Anda sebagai masukan RegisterUsage atau dengan MeterUsage cara yang tidak dapat dimodifikasi oleh pembeli. Namun, jika produk Anda mengelola kode produk dengan cara yang dapat ditimpa pelanggan, seperti bagan Helm AWS CloudFormation, atau manifes Kubernetes, Anda harus mempertahankan daftar kode produk tepercaya. AWS Marketplace Ini untuk memastikan bahwa kode produk yang dilewati perangkat lunak Anda sebagai input RegisterUsage atau MeterUsage valid.
- Jika salah satu kode produk tepercaya Anda adalah produk gratis, pastikan kode produk tersebut tidak dapat digunakan sebagai pengganti kode produk berbayar.

Harga kontrak dengan AWS License Manager

Untuk produk berbasis kontainer dengan harga kontrak, Anda gunakan AWS License Manager untuk mengaitkan lisensi dengan produk Anda.

AWS License Manager adalah alat manajemen lisensi yang memungkinkan aplikasi Anda untuk melacak dan memperbarui lisensi (juga dikenal sebagai hak) yang telah dibeli oleh pelanggan. Bagian ini memberikan informasi tentang cara mengintegrasikan produk Anda AWS License Manager. Setelah integrasi selesai, Anda dapat mempublikasikan daftar produk Anda AWS Marketplace.

Untuk informasi selengkapnya AWS License Manager, lihat [Panduan AWS License Manager Pengguna](#) dan [AWS License Manager](#) bagian Referensi AWS CLI Perintah.

Note

- Pelanggan tidak dapat meluncurkan instance baru dari kontainer setelah masa berakhirnya kontrak. Namun, selama durasi kontrak, mereka dapat meluncurkan sejumlah instance.

Lisensi ini tidak terikat pada node atau instance tertentu. Perangkat lunak apa pun yang berjalan pada wadah apa pun pada node apa pun dapat memeriksa lisensi selama memiliki AWS kredensialnya yang ditetapkan.

- Pembuatan Penawaran Pribadi — Penjual dapat menghasilkan penawaran pribadi untuk produk menggunakan alat pembuatan penawaran Pribadi di Portal Manajemen AWS Marketplace.
- Pelaporan — Anda dapat mengatur umpan data dengan menyiapkan bucket Amazon S3 di bagian Laporan di bagian. Portal Manajemen AWS Marketplace Untuk informasi selengkapnya, lihat [Laporan penjual, umpan data, dan dasbor di AWS Marketplace](#).

Alur kerja integrasi

Langkah-langkah berikut menunjukkan alur kerja untuk mengintegrasikan produk kontainer Anda dengan: AWS License Manager

1. Penjual menciptakan produk dengan AWS License Manager integrasi.
2. Penjual mencantumkan produk di AWS Marketplace.
3. Pembeli menemukan produk AWS Marketplace dan membelinya.
4. Lisensi dikirim ke pembeli di dalamnya Akun AWS.
5. Pembeli menggunakan perangkat lunak dengan meluncurkan EC2 instans Amazon, tugas Amazon ECS, atau perangkat lunak pod Amazon EKS. Pelanggan menyebarkan menggunakan peran IAM.
6. Perangkat lunak membaca lisensi di AWS License Manager akun pembeli, menemukan hak yang dibeli, dan menyediakan fitur yang sesuai.

Note

License Manager tidak melakukan pelacakan atau pembaruan apa pun; ini dilakukan oleh aplikasi penjual.

Mengkonfigurasi pengukuran per jam dengan AWS Marketplace Metering Service

Note

Untuk penerapan Amazon EKS, perangkat lunak Anda harus menggunakan [peran IAM untuk akun layanan \(IRSA\)](#) untuk menandatangani panggilan API untuk operasi API. [RegisterUsage](#) Menggunakan [EKS Pod Identity](#), peran node, atau kunci akses jangka panjang tidak didukung.

Untuk penerapan Amazon ECS, perangkat lunak Anda harus menggunakan peran [IAM tugas Amazon ECS](#) untuk menandatangani panggilan API untuk operasi API. [RegisterUsage](#) Menggunakan peran node atau kunci akses jangka panjang tidak didukung.

Jika produk kontainer Anda menggunakan harga per jam per tugas atau per-pod alih-alih dimensi harga terukur khusus, Anda tidak perlu menentukan dimensi pengukuran khusus. Anda dapat menggunakan AWS Marketplace Metering Service untuk pengukuran per jam dengan produk kontainer. AWS Marketplace Bagian berikut menunjukkan cara mengonfigurasi pengukuran per jam dengan AWS Marketplace Metering Service.

Perangkat lunak pengukur operasi RegisterUsage API digunakan per tugas Amazon Elastic Container Service (Amazon ECS) atau per pod Amazon Elastic Kubernetes Service (Amazon EKS), per jam, dengan penggunaan prorata ke yang kedua. Penggunaan minimal 1 menit berlaku untuk tugas atau pod yang berumur pendek. Pengukuran berkelanjutan untuk penggunaan perangkat lunak secara otomatis ditangani oleh AWS Marketplace Metering Control Plane. Perangkat lunak Anda tidak diperlukan untuk melakukan tindakan khusus pengukuran apa pun kecuali menelepon RegisterUsage sekali untuk pengukuran penggunaan perangkat lunak untuk memulai.

RegisterUsage harus segera dipanggil pada saat meluncurkan wadah. Jika Anda tidak mendaftarkan kontainer dalam 6 jam pertama peluncuran kontainer, AWS Marketplace Metering Service tidak memberikan jaminan pengukuran apa pun untuk bulan-bulan sebelumnya. Namun, pengukuran akan berlanjut untuk bulan berjalan ke depan sampai wadah berakhir.

The AWS Marketplace Metering Control Plane terus menagih pelanggan untuk menjalankan tugas Amazon ECS dan pod Amazon EKS, terlepas dari status langganan pelanggan. Ini menghilangkan kebutuhan perangkat lunak Anda untuk melakukan pemeriksaan hak setelah peluncuran tugas atau pod awal yang berhasil.

Untuk informasi selengkapnya tentang mengintegrasikan AWS Marketplace Metering Service API dengan produk kontainer dengan harga per jam, lihat lab [Integrasikan dengan pengukuran per jam](#) di bengkel penjual.AWS Marketplace

Topik

- [Prasyarat pengukuran per jam](#)
- [Integrasi pengujian untuk RegisterUsage](#)
- [Penanganan kesalahan untuk RegisterUsage](#)
- [Mengintegrasikan produk container Anda dengan AWS Marketplace Metering Service menggunakan AWS SDK untuk Java](#)

Prasyarat pengukuran per jam

Sebelum menerbitkan produk, Anda harus melakukan hal berikut:

1. Buat produk kontainer baru di Portal Manajemen AWS Marketplace, dan catat kode produknya.

Untuk informasi selengkapnya, lihat [Ikhtisar: Buat produk kontainer](#).

2. Gunakan peran AWS Identity and Access Management (IAM) untuk tugas atau pod yang menjalankan aplikasi Anda dengan izin IAM yang diperlukan untuk memanggil RegisterUsage Kebijakan terkelola AWSMarketplaceMeteringRegisterUsage meliputi izin-izin ini. Untuk informasi selengkapnya tentang kebijakan, lihat [AWSMarketplaceMeteringFullAccess](#) di Referensi Kebijakan AWS Terkelola.
3. (Opsional) Jika Anda ingin melihat logging, kami sarankan Anda mengaktifkan AWS CloudTrail logging dalam tugas atau definisi pod.
4. Lakukan panggilan uji ke operasi RegisterUsage API dengan catatan untuk semua dimensi harga yang Anda tetapkan.

Integrasi pengujian untuk **RegisterUsage**

Gunakan operasi RegisterUsage API untuk menguji integrasi Anda sebelum mengirimkan gambar Anda AWS Marketplace untuk dipublikasikan.

Panggil RegisterUsage dari gambar kontainer dengan menjalankan produk Anda di Amazon ECS atau Amazon EKS. Gunakan AWS akun yang Anda gunakan untuk membuat daftar produk AWS Marketplace. Integrasi pengukuran Anda harus secara dinamis mengatur Wilayah AWS, bukan

hardcoding itu. Namun, saat menguji, luncurkan setidaknya satu tugas Amazon ECS atau pod Amazon EKS yang berisi kontainer berbayar Anda di Wilayah AS Timur (Virginia Utara). Dengan melakukan ini, tim AWS Marketplace operasi dapat memverifikasi pekerjaan Anda dengan log di Wilayah tersebut.

Note

Jika produk Anda mendukung Amazon ECS dan Amazon EKS, Anda hanya perlu meluncurkan di Amazon EKS bagi kami untuk memvalidasi integrasi Anda.

Anda tidak dapat sepenuhnya menguji integrasi sampai produk Anda dipublikasikan dengan semua metadata dan informasi harga yang diperlukan. Jika diminta, tim operasi AWS Marketplace katalog dapat memverifikasi penerimaan catatan pengukuran Anda.

Penanganan kesalahan untuk **RegisterUsage**

Jika image container Anda terintegrasi dengan AWS Marketplace Metering Service dan menerima pengecualian selain `ThrottlingException` saat startup container, Anda harus menghentikan container untuk mencegah penggunaan yang tidak sah.

Pengecualian selain `ThrottlingException` dilemparkan hanya pada panggilan awal ke operasi `RegisterUsage` API. Panggilan berikutnya dari tugas Amazon ECS yang sama atau pod Amazon EKS tidak dibuang `CustomerNotSubscribedException` meskipun pelanggan berhenti berlangganan saat tugas atau pod masih berjalan. Pelanggan ini masih dikenakan biaya untuk menjalankan kontainer setelah mereka berhenti berlangganan, dan penggunaannya dilacak.

Tabel berikut menjelaskan kesalahan yang mungkin ditimbulkan oleh operasi `RegisterUsage` API. Setiap bahasa pemrograman AWS SDK memiliki seperangkat pedoman penanganan kesalahan yang dapat Anda rujuk untuk informasi tambahan.

Kesalahan	Deskripsi
<code>InternalServerErrorException</code>	<code>RegisterUsage</code> tidak tersedia.
<code>CustomerNotEntitledException</code>	Pelanggan tidak memiliki langganan yang valid untuk produk tersebut.

Kesalahan	Deskripsi
<code>InvalidProductCodeException</code>	<code>ParameterProductCode</code> nilai yang dilewatkan sebagai bagian dari permintaan tidak ada.
<code>InvalidPublicKeyException</code>	<code>ParameterPublicKeyVersion</code> nilai yang dilewatkan sebagai bagian dari permintaan tidak ada.
<code>PlatformNotSupportedException</code>	AWS Marketplace tidak mendukung penggunaan pengukuran dari platform yang mendasarinya. Hanya Amazon ECS, Amazon EKS, dan AWS Fargate yang didukung.
<code>ThrottlingException</code>	Panggilan ke <code>RegisterUsage</code> diperlambat.
<code>InvalidRegionException</code>	<code>RegisterUsage</code> harus dipanggil sama Wilayah AWS dengan tugas Amazon ECS atau pod Amazon EKS diluncurkan. Hal ini mencegah kontainer dari memilih Wilayah (misalnya, <code>withRegion("us-east-1")</code>) saat memanggil <code>RegisterUsage</code> .

Mengintegrasikan produk container Anda dengan AWS Marketplace Metering Service menggunakan AWS SDK untuk Java

Anda dapat menggunakan AWS SDK untuk Java untuk berintegrasi dengan AWS Marketplace Metering Service. Pengukuran berkelanjutan untuk penggunaan perangkat lunak secara otomatis ditangani oleh AWS Marketplace Metering Control Plane. Perangkat lunak Anda tidak diperlukan untuk melakukan tindakan khusus pengukuran apa pun kecuali menelepon `RegisterUsage` sekali untuk pengukuran penggunaan perangkat lunak untuk memulai. Topik ini memberikan contoh implementasi menggunakan AWS SDK untuk Java untuk mengintegrasikan dengan `RegisterUsage` tindakan [Layanan AWS Marketplace Pengukuran](#).

`RegisterUsage` harus segera dipanggil pada saat meluncurkan wadah. Jika Anda tidak mendaftarkan kontainer dalam 6 jam pertama peluncuran kontainer, AWS Marketplace Metering

Service tidak memberikan jaminan pengukuran apa pun untuk bulan-bulan sebelumnya. Namun, pengukuran akan berlanjut untuk bulan berjalan ke depan sampai wadah berakhir.

Untuk kode sumber lengkap, lihat [RegisterUsage Contoh Java](#). Banyak dari langkah-langkah ini berlaku terlepas dari bahasa AWS SDK.

Contoh langkah untuk integrasi AWS Marketplace Metering Service

1. Masuk ke [Portal Manajemen AWS Marketplace](#).
2. Dari Aset Pilih Kontainer untuk mulai membuat produk kontainer baru. Membuat produk menghasilkan kode produk untuk produk untuk mengintegrasikan dengan citra kontainer Anda. Untuk informasi tentang izin IAM yang diperlukan, lihat [the section called “AWS Marketplace izin API pengukuran dan hak”](#).
3. Unduh [AWS Java SDK](#) publik.

Important

Untuk memanggil metering APIs dari Amazon EKS, Anda harus [menggunakan AWS SDK yang didukung](#) dan berjalan di kluster Amazon EKS yang menjalankan Kubernetes 1.13 atau yang lebih baru.

4. (Opsional) Jika Anda mengintegrasikan dengan RegisterUsage tindakan dan ingin melakukan verifikasi tanda tangan digital, Anda perlu mengonfigurasi pustaka verifikasi [BouncyCastle](#) tanda tangan di classpath aplikasi Anda.

Jika Anda ingin menggunakan JSON Web Token (JWT), Anda juga harus menyertakan [Java JWT](#) perpustakaan di classpath aplikasi Anda. Menggunakan JWT memberikan pendekatan yang lebih sederhana untuk verifikasi tanda tangan tetapi tidak diperlukan, dan Anda dapat menggunakan mandiri sebagai BouncyCastle gantinya. Baik Anda menggunakan JWT atau BouncyCastle, Anda perlu menggunakan sistem build seperti Maven untuk menyertakan dependensi transitif atau JWT di classpath aplikasi Anda. BouncyCastle

```
// Required for signature verification using code sample
<dependency>
  <groupId>org.bouncycastle</groupId>
  <artifactId>bcpkix-jdk15on</artifactId>
  <version>1.60</version>
</dependency>
```

```
// This one is only required for JWT
<dependency>
  <groupId>com.nimbusds</groupId>
  <artifactId>nimbus-jose-jwt</artifactId>
  <version>6.0</version>
</dependency>
```

5. Panggilan `RegisterUsage` dari setiap citra kontainer berbayar dalam penawaran produk Anda. `ProductCode` dan `PublicKeyVersion` diperlukan parameter, dan semua input lainnya adalah opsional. Berikut ini adalah contoh muatan untuk `RegisterUsage`.

```
{
  "ProductCode" : "string", // (required)
  "PublicKeyVersion": 1,    // (required)
  "Nonce": "string",       // (optional) to scope down the registration
                           // to a specific running software
                           // instance and guard against
                           // replay attacks
}
```

Note

Dimungkinkan untuk melihat masalah sementara dalam menghubungkan ke AWS Marketplace Metering Service. AWS Marketplace sangat merekomendasikan untuk menerapkan percobaan ulang hingga 30 menit, dengan mundur eksponensial, untuk menghindari pemadaman jangka pendek atau masalah jaringan.

6. `RegisterUsage` menghasilkan tanda tangan digital RSA-PSS menggunakan SHA-256 yang dapat Anda gunakan untuk memverifikasi keaslian permintaan. Tanda tangan meliputi kolom-kolom berikut: `ProductCode`, `PublicKeyVersion`, dan `Nonce`. Untuk memverifikasi tanda tangan digital, Anda harus mempertahankan bidang ini dari permintaan. Kode berikut adalah contoh respons terhadap `RegisterUsage` Panggilan.

```
{
  "Signature": "<<JWT Token>>"
}

// Where the JWT Token is composed of 3 dot-separated,
// base-64 URL Encoded sections.
// e.g. eyJhbGcVCj9.eyJzdWIMzkwMjJ9.rr09Qw0SXRWTe
```

```
// Section 1: Header/Algorithm
{
  "alg": "PS256",
  "typ": "JWT"
}

// Section 2: Payload
{
  "ProductCode" : "string",
  "PublicKeyVersion": 1,
  "Nonce": "string",
  "iat": date // JWT issued at claim
}

// Section 3: RSA-PSS SHA256 signature
"rr09Q4FEi3gweH3X4lrt2okf5zwIatUUwERlw016wTy_21Nv8S..."
```

7. Buat kembali versi baru gambar penampung Anda yang menyertakan `RegisterUsage` panggilan, beri tag penampung, dan dorong ke registri kontainer apa pun yang kompatibel dengan Amazon ECS atau Amazon EKS, seperti Amazon ECR atau Amazon ECR Public. Jika Anda menggunakan Amazon ECR, memastikan bahwa akun meluncurkan tugas Amazon ECS atau Amazon EKS pod memiliki izin pada repositori Amazon ECR. Jika tidak, peluncuran gagal.
8. Buat [IAM](#) peran yang memberikan izin untuk kontainer Anda untuk menelepon `RegisterUsage`, seperti yang didefinisikan dalam kode berikut. Anda harus menyediakan IAM role ini di parameter [Peran tugas](#) Amazon ECS atau definisi pod Amazon EKS.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "aws-marketplace:RegisterUsage"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

9. Buat tugas Amazon ECS atau definisi pod Amazon EKS yang mereferensikan wadah yang telah terintegrasi AWS Marketplace dan mereferensikan peran IAM yang Anda buat di langkah 7. Anda harus mengaktifkan AWS CloudTrail logging dalam definisi tugas jika Anda ingin melihat logging.
10. Buat Amazon ECS atau Amazon EKS cluster untuk melaksanakan tugas Anda atau pod. Untuk informasi selengkapnya tentang cara membuat Amazon ECS cluster, lihat [Membuat Gugus](#) di Panduan Pengembang Layanan Elastic kontainer. Untuk informasi selengkapnya tentang membuat cluster Amazon EKS (menggunakan Kubernetes versi 1.1.3.x atau yang lebih baru), lihat [Membuat klaster EKS Amazon](#).
11. Konfigurasi cluster Amazon ECS atau Amazon EKS dan luncurkan definisi tugas Amazon ECS atau pod Amazon EKS yang Anda buat, di us-east-1. Wilayah AWS Hanya selama proses pengujian ini, sebelum produk ditayangkan, Anda harus menggunakan wilayah ini.
12. Saat Anda mendapatkan responss yang valid kembali dari `RegisterUsage`, Anda dapat mulai membuat produk kontainer Anda. Untuk pertanyaan, hubungi [AWS Marketplace Operasi Penjual](#) Tim.

RegisterUsage Contoh Java

Contoh berikut menggunakan AWS SDK untuk Java dan AWS Marketplace Metering Service untuk memanggil `RegisterUsage` operasi. Verifikasi tanda tangan bersifat opsional, tetapi jika Anda ingin melakukan verifikasi tanda tangan, Anda harus menyertakan pustaka verifikasi tanda tangan digital yang diperlukan. Contoh ini hanya untuk tujuan ilustrasi.

```
import com.amazonaws.auth.PEM;
import com.amazonaws.services.marketplacemetering.AWSMarketplaceMetering;
import com.amazonaws.services.marketplacemetering.AWSMarketplaceMeteringClientBuilder;
import com.amazonaws.services.marketplacemetering.model.RegisterUsageRequest;
import com.amazonaws.services.marketplacemetering.model.RegisterUsageResult;
import com.amazonaws.util.json.Jackson;
import com.fasterxml.jackson.databind.JsonNode;
import com.nimbusds.jose.JWSObject;
import com.nimbusds.jose.JWSVerifier;
import com.nimbusds.jose.crypto.RSASSAVerifier;
import java.io.ByteArrayInputStream;
import java.nio.charset.StandardCharsets;
import java.security.PublicKey;
import java.security.Security;
import java.security.Signature;
import java.security.interfaces.RSAPublicKey;
```

```
import java.util.Base64;
import java.util.Optional;
import java.util.UUID;
import org.bouncycastle.jce.provider.BouncyCastleProvider;

/**
 * Class for making calls out to &MKT; Metering Service.
 */
class RegisterUsage {

    private static final String PRODUCT_CODE = ".....";

    private final AWSMarketplaceMetering registerUsageClient;
    private final SignatureVerifier signatureVerifier;
    private final int publicKeyVersion;

    public RegisterUsage(final SignatureVerifier signatureVerifier) {
        this.signatureVerifier = signatureVerifier;
        this.publicKeyVersion = PublicKeyProvider.PUBLIC_KEY_VERSION;
        this.registerUsageClient =
AWSMarketplaceMeteringClientBuilder.standard().build();
    }

    /**
     * Shows how to call RegisterUsage client and verify digital signature.
     */
    public void callRegisterUsage() {
        RegisterUsageRequest request = new RegisterUsageRequest()
            .withProductCode(PRODUCT_CODE)
            .withPublicKeyVersion(publicKeyVersion)
            .withNonce(UUID.randomUUID().toString());

        // Execute call to RegisterUsage (only need to call once at container startup)
        RegisterUsageResult result = this.registerUsageClient.registerUsage(request);

        // Verify Digital Signature w/o JWT
        boolean isSignatureValid = this.signatureVerifier.verify(request, result);
        if (!isSignatureValid) {
            throw new RuntimeException("Revoke entitlement, digital signature
invalid.");
        }
    }
}
```

```
/**
 * Signature verification class with both a JWT-library based verification
 * and a non-library based implementation.
 */
class SignatureVerifier {
    private static BouncyCastleProvider BC = new BouncyCastleProvider();

    private static final String SIGNATURE_ALGORITHM = "SHA256withRSA/PSS";

    private final PublicKey publicKey;

    public SignatureVerifier(PublicKeyProvider publicKeyProvider) {
        this.publicKey = publicKeyProvider.getPublicKey().orElse(null);
        Security.addProvider(BC);
    }

    /**
     * Example signature verification using the NimbusJOSEJWT library to verify the JWT
     * Token.
     *
     * @param request RegisterUsage Request.
     * @param result RegisterUsage Result.
     * @return true if the token matches.
     */
    public boolean verifyUsingNimbusJOSEJWT(final RegisterUsageRequest request, final
RegisterUsageResult result) {
        if (!getPublicKey().isPresent()) {
            return false;
        }

        try {
            JWSVerifier verifier = new RSASSAVerifier((RSAPublicKey)
getPublicKey().get());
            JWSObject jwsObject = JWSObject.parse(result.getSignature());
            return jwsObject.verify(verifier) &&
validatePayload(jwsObject.getPayload().toString(), request, result);
        } catch (Exception e) {
            // log error
            return false;
        }
    }

    /**
     * Example signature verification without any JWT library support.
     */
}
```

```
*
* @param request RegisterUsage Request.
* @param result RegisterUsage Result.
* @return true if the token matches.
*/
public boolean verify(final RegisterUsageRequest request, final RegisterUsageResult
result) {
    if (!getPublicKey().isPresent()) {
        return false;
    }
    try {
        String[] jwtParts = result.getSignature().split("\\.");
        String header = jwtParts[0];
        String payload = jwtParts[1];
        String payloadSignature = jwtParts[2];

        Signature signature = Signature.getInstance(SIGNATURE_ALGORITHM, BC);
        signature.initVerify(getPublicKey().get());
        signature.update(String.format("%s.%s", header,
payload).getBytes(StandardCharsets.UTF_8));
        boolean verified = signature.verify(Base64.getUrlDecoder()
            .decode(payloadSignature.getBytes(StandardCharsets.UTF_8)));

        String decodedPayload = new String(Base64.getUrlDecoder().decode(payload));
        return verified && validatePayload(decodedPayload, request, result);
    } catch (Exception e) {
        // log error
        return false;
    }
}

/**
 * Validate each value in the returned payload matches values originally
 * supplied in the request to RegisterUsage. TimeToLiveInMillis and
 * PublicKeyExpirationTimestamp will have the values in the payload compared
 * to values in the signature
 */
private boolean validatePayload(final String payload, final RegisterUsageRequest
request,
                                final RegisterUsageResult result) {
    try {
        JsonNode payloadJson = Jackson.getObjectMapper().readTree(payload);
        boolean matches = payloadJson.get("productCode")
            .asText()
```

```

        .equals(request.getProductCode());
    matches = matches && payloadJson.get("nonce")
        .asText()
        .equals(request.getNonce());
    return matches = matches && payloadJson.get("publicKeyVersion")
        .asText()
        .equals(String.valueOf(request.getPublicKeyVersion()));

    } catch (Exception ex) {
        // log error
        return false;
    }
}

private Optional<PublicKey> getPublicKey() {
    return Optional.ofNullable(this.publicKey);
}
}

/**
 * Public key provider taking advantage of the &AWS; PEM Utility.
 */
class PublicKeyProvider {
    // Replace with your public key. Ensure there are new-lines ("\n") in the
    // string after "-----BEGIN PUBLIC KEY-----\n" and before "\n-----END PUBLIC
    KEY-----".
    private static final String PUBLIC_KEY =
        "-----BEGIN PUBLIC KEY-----\n"
        + "MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQDdlatRjRjogo3WojgGHFHYLugd
\n"
        + "UWAY9iR3fy4arWNA1KoS8kVw33cJibXr8bvWUAUparCwlvdbH6dvE0fou0/gCFQs
\n"
        + "HUfQrSDv+MuSUMAe8jzKE4qW+jK+xQU9a03GUnKHkk1e+Q0pX/g6jXZ7r1/xAK5D
\n"
        + "o2kQ+X5xK9cipRgEKwIDAQAB\n"
        + "-----END PUBLIC KEY-----";

    public static final int PUBLIC_KEY_VERSION = 1;

    public Optional<PublicKey> getPublicKey() {
        try {
            return Optional.of(PEM.readPublicKey(new ByteArrayInputStream(
                PUBLIC_KEY.getBytes(StandardCharsets.UTF_8))));
        } catch (Exception e) {

```

```
        // log error
        return Optional.empty();
    }
}
```

Mengkonfigurasi pengukuran khusus untuk produk kontainer dengan AWS Marketplace Metering Service

Note

Untuk penerapan Amazon EKS, perangkat lunak Anda harus menggunakan [peran IAM untuk akun layanan \(IRSA\)](#) untuk menandatangani panggilan API untuk operasi API. [MeterUsage](#) Menggunakan [EKS Pod Identity](#), peran node, atau kunci akses jangka panjang tidak didukung.

Untuk penerapan Amazon ECS, perangkat lunak Anda harus menggunakan peran [IAM tugas Amazon ECS](#) untuk menandatangani panggilan API untuk operasi API. [MeterUsage](#) Menggunakan peran node atau kunci akses jangka panjang tidak didukung.

AWS Marketplace produk kontainer dapat memiliki pengukuran khusus hingga 24 dimensi harga yang berbeda per produk. Setiap dimensi dapat memiliki harga kontrak jangka panjang yang terkait dengannya. Untuk mengaktifkan pengukuran khusus, integrasikan produk container Anda dengan AWS Marketplace Metering Service. Anda dapat menentukan unit harga Anda sendiri dan pengukuran kustom untuk penggunaan tersebut AWS untuk penagihan menggunakan operasi [MeterUsage](#) API. Bagian berikut menunjukkan cara mengonfigurasi pengukuran khusus untuk produk kontainer Anda.

Dimensi harga didefinisikan di dua lokasi, sekali saat membuat produk Anda di Portal Manajemen AWS Marketplace (portal penjual) dan sekali di perangkat lunak Anda untuk melakukan `MeterUsage` operasi. Metode dua faktor ini memastikan bahwa penawaran berikutnya bekerja sebagaimana dimaksud sebelum dibuat tersedia untuk umum.

Untuk mengatur pengukuran khusus, Anda harus memilih kategori penggunaan, tipe unit, dan dimensi harga:

- **Kategori penggunaan**— Kategori penggunaan membantu pembeli memahami produk Anda dan cara menggunakannya.

- Tipe unit— Tipe unit mendefinisikan satuan ukuran untuk penagihan. Misalnya, bandwidth diukur dalam GBps atau MBps, jumlah host, atau data yang diukur dalam MB, GB, atau TB.
- Dimensi harga — Dimensi harga mewakili fitur atau layanan yang telah Anda tetapkan harga per unit (misalnya, pengguna, pemindaian, vCPUs, atau agen yang digunakan). Dimensi harga bersifat publik. Namun, Anda masih dapat menentukan penawaran pribadi dan Bring Your Own License (BYOL) untuk produk publik. Jangan mengirim harga dalam catatan pengukuran. Anda mengukur jumlah unit, dan kami menggunakannya bersama dengan harga yang Anda tentukan saat membuat produk Anda untuk menghitung tagihan pembeli.

Jika harga produk Anda tidak sesuai dengan kategori atau tipe unit yang telah ditentukan sebelumnya, Anda dapat memilih kategori Unit generik. Kemudian, gunakan deskripsi dimensi untuk menggambarkan apa unitnya.

Opsional, Anda dapat mendistribusikan penggunaan ke alokasi oleh properti yang Anda melacak. Alokasi direpresentasikan sebagai tag kepada pembeli. Tag ini memungkinkan pembeli untuk melihat biaya mereka dibagi menjadi penggunaan berdasarkan nilai tag. Sebagai contoh, jika Anda mengisi oleh pengguna, dan pengguna memiliki properti “Departemen”, Anda dapat membuat alokasi penggunaan dengan tag yang memiliki kunci “Departemen”, dan satu alokasi per nilai. Ini tidak mengubah harga, dimensi, atau total penggunaan yang Anda laporkan, tetapi memungkinkan pelanggan Anda untuk melihat biaya mereka berdasarkan kategori yang sesuai dengan produk Anda.

Kami menyarankan Anda mengirim catatan pengukuran setiap jam. Namun, Anda dapat menggabungkan penggunaan selama periode harian atau bulanan juga. Jika Anda mengalami pemadaman, Anda dapat menggabungkan penggunaan perangkat lunak pembeli dan mengirimkannya dalam pengukuran jam berikutnya. Anda tidak dapat mengirim lebih dari satu catatan per jam.

Untuk informasi selengkapnya tentang mengintegrasikan AWS Marketplace Metering Service API untuk produk kontainer dengan harga pengukuran khusus, lihat lab [Integrasikan dengan pengukuran khusus](#) bengkel penjual.AWS Marketplace

Important

Percobaan gratis dan hak prabayar dilacak pada tingkat per jam. Akibatnya, mengirim catatan ini secara terpisah dapat menyebabkan pembeli menjadi kelebihan biaya.

Topik

- [Prasyarat pengukuran](#)
- [MeterUsageIntegrasi pengujian](#)
- [Penanganan kesalahan untuk MeterUsage](#)
- [Penandaan terukur vendor \(Opsional\)](#)
- [Contoh kode](#)
- [Mengintegrasikan produk kontainer Anda menggunakan pengukuran khusus dengan dan AWS Marketplace Metering Service AWS SDK untuk Java](#)

Prasyarat pengukuran

Sebelum menerbitkan produk, Anda harus melakukan hal berikut:

1. Buat produk kontainer baru di Portal Manajemen AWS Marketplace, dan catat kode produknya.
2. Gunakan peran AWS Identity and Access Management (IAM) untuk tugas atau pod yang menjalankan aplikasi Anda dengan izin IAM yang diperlukan untuk memanggil `MeterUsage`. Kebijakan terkelola `AWSMarketplaceMeteringRegisterUsage` meliputi izin-izin ini. Untuk informasi selengkapnya tentang kebijakan, lihat [AWSMarketplaceMeteringFullAccess](#) di Referensi Kebijakan AWS Terkelola.
3. (Opsional) Kami menyarankan Anda mengaktifkan AWS CloudTrail logging dalam tugas atau definisi pod jika Anda ingin melihat logging.
4. Lakukan panggilan uji ke operasi `MeterUsage` API dengan catatan untuk semua dimensi harga yang Anda tetapkan.

MeterUsageIntegrasi pengujian

Gunakan `MeterUsage` operasi untuk menguji integrasi Anda sebelum mengirimkan gambar Anda AWS Marketplace untuk dipublikasikan.

Panggilan `MeterUsage` dari gambar kontainer dengan menjalankan produk Anda di Amazon Elastic Container Service (Amazon ECS) Service (Amazon ECS) atau Amazon Elastic Kubernetes Service (Amazon EKS Akun AWS) dengan yang Anda gunakan untuk membuat daftar produk. AWS Marketplace Integrasi pengukuran Anda harus secara dinamis mengatur Wilayah AWS, bukan hard coding itu. Namun, saat menguji, luncurkan setidaknya satu tugas Amazon ECS atau pod Amazon EKS yang berisi kontainer berbayar Anda di Wilayah AS Timur (Virginia Utara) sehingga tim AWS Marketplace operasi dapat memverifikasi pekerjaan Anda dengan log di Wilayah tersebut.

Note

- Jika produk Anda mendukung Amazon ECS dan Amazon EKS, Anda hanya perlu meluncurkan di Amazon EKS bagi kami untuk memvalidasi integrasi Anda.
- Uji setiap dimensi sebelum meluncurkan produk Anda ke publik dan setelah menambahkan dimensi baru. Jika Anda tidak mengirim catatan pengukuran untuk setiap dimensi yang terkait dengan produk kontainer, itu akan mengakibatkan kesalahan dengan permintaan yang gagal.

Anda tidak dapat sepenuhnya menguji integrasi sampai produk Anda dipublikasikan dengan semua metadata dan informasi harga yang diperlukan. Jika diminta, tim operasi AWS Marketplace katalog dapat memverifikasi penerimaan catatan pengukuran Anda.

Penanganan kesalahan untuk `MeterUsage`

Jika citra kontainer Anda terintegrasi dengan `MeterUsage` operasi dan menerima pengecualian selain `ThrottlingException` pada startup kontainer, Anda harus mengakhiri kontainer untuk mencegah penggunaan yang tidak sah.

Pengecualian selain `ThrottlingException` dilemparkan hanya pada panggilan awal untuk `MeterUsage`. Panggilan berikutnya dari tugas Amazon ECS yang sama atau Amazon EKS pod tidak membuang `CustomerNotSubscribedException` bahkan jika pelanggan berhenti berlangganan sementara tugas atau pod masih berjalan. Pelanggan ini masih dikenakan biaya untuk menjalankan kontainer setelah mereka berhenti berlangganan dan penggunaannya dilacak.

Lihat [MeterUsage](#) di Referensi AWS Marketplace Metering Service API untuk deskripsi rinci tentang kesalahan umum untuk `MeterUsage`. Setiap bahasa pemrograman AWS SDK memiliki seperangkat pedoman penanganan kesalahan yang dapat Anda rujuk untuk informasi tambahan.

Penandaan terukur vendor (Opsional)

Penandaan yang diukur vendor membantu Vendor Perangkat Lunak Independen (ISVs) memberi pembeli wawasan yang lebih terperinci tentang penggunaan perangkat lunak mereka dan dapat membantu mereka melakukan alokasi biaya.

Ada banyak cara untuk menandai penggunaan perangkat lunak pembeli. Salah satu caranya adalah dengan terlebih dahulu bertanya kepada pembeli Anda apa yang ingin mereka lihat dalam alokasi

biaya mereka. Kemudian Anda dapat membagi penggunaan di seluruh properti yang Anda lacak untuk akun pembeli. Contoh properti termasuk `AccountId`, `Business UnitCost Centers`, dan metadata relevan lainnya untuk produk Anda. Properti ini diekspos ke pembeli sebagai tag. Dengan menggunakan tag, pembeli dapat melihat biayanya dibagi menjadi penggunaan berdasarkan nilai tag di Konsol AWS Penagihan (<https://console.aws.amazon.com/costmanagement/>). Penandaan yang diukur vendor tidak mengubah harga, dimensi, atau total penggunaan yang Anda laporkan. Hal ini memungkinkan pelanggan Anda untuk melihat biaya mereka berdasarkan kategori yang sesuai dengan produk Anda.

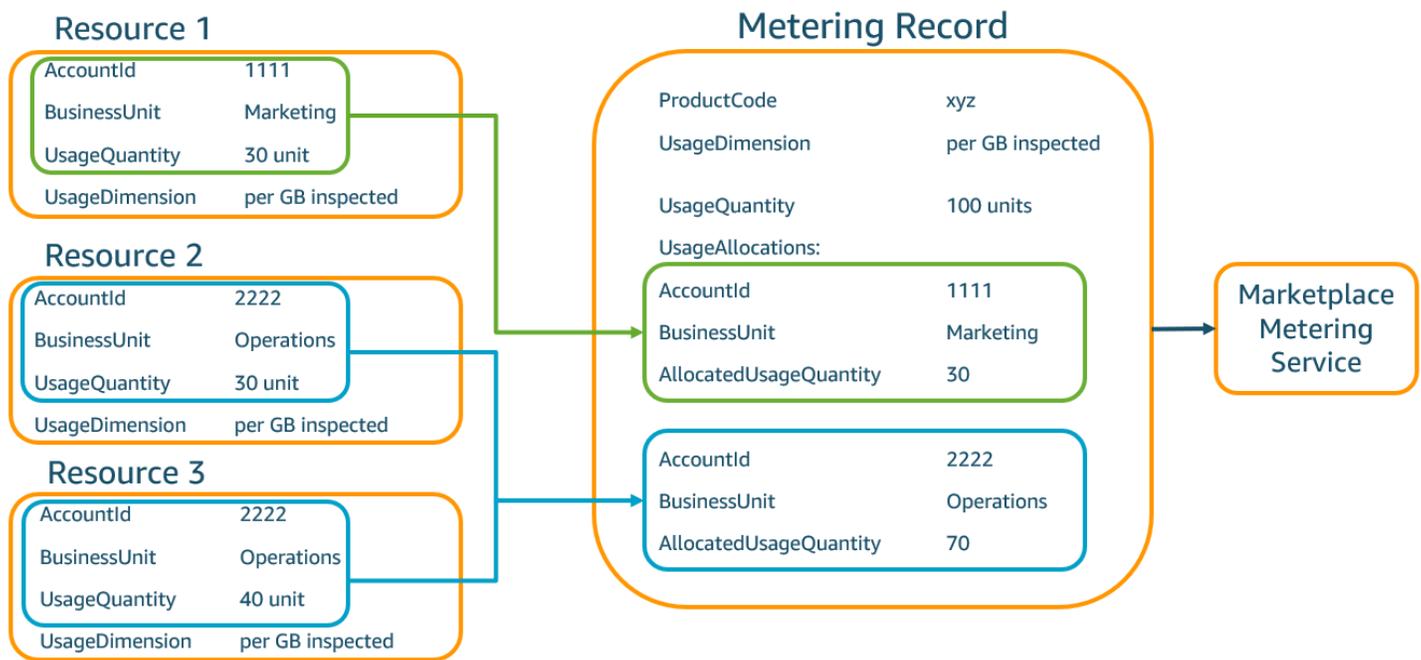
Dalam kasus penggunaan umum, pembeli berlangganan produk Anda dengan satu Akun AWS. Pembeli juga memiliki banyak pengguna yang terkait dengan langganan produk yang sama. Anda dapat membuat alokasi penggunaan dengan tag yang memiliki kunci `AccountId`, dan kemudian mengalokasikan penggunaan untuk setiap pengguna. Dalam hal ini, pembeli dapat mengaktifkan `AccountId` tag di konsol Billing and Cost Management mereka dan menganalisis penggunaan pengguna individu.

Pengalaman penjual

Penjual dapat menggabungkan catatan pengukuran untuk sumber daya dengan kumpulan tag yang sama alih-alih menggabungkan penggunaan untuk semua sumber daya. Misalnya, penjual dapat membuat catatan pengukuran yang mencakup ember yang berbeda dari `UsageAllocations`. Setiap bucket mewakili `UsageQuantity` satu set tag, seperti `AccountId` dan `BusinessUnit`.

Dalam diagram berikut, Resource 1 memiliki satu set `AccountId` dan `BusinessUnit` tag yang unik, dan muncul di Rekaman Pengukuran sebagai entri tunggal.

Resource 2 dan Resource 3 keduanya memiliki `AccountId` tag yang sama `2222`, dan `BusinessUnit` tag yang sama `Operations`. Akibatnya, mereka digabungkan menjadi satu `UsageAllocations` entri dalam Rekaman Pengukuran.



Penjual juga dapat menggabungkan sumber daya tanpa tag menjadi satu UsageAllocation dengan jumlah penggunaan yang dialokasikan dan mengirimkannya sebagai salah satu entri di UsageAllocations

Batasan meliputi:

- Jumlah tag — 5
- Ukuran UsageAllocations (kardinalitas) — 2.500

Validasi meliputi:

- Karakter diperbolehkan untuk kunci tag dan nilai — a-zA-Z 0-9+ -= . _:\ /@
- Tag maksimum di seluruh UsageAllocation daftar - 5
- Dua tidak UsageAllocations dapat memiliki tag yang sama (yaitu, kombinasi yang sama dari kunci tag dan nilai). Jika itu masalahnya, mereka harus menggunakan hal yang samaUsageAllocation.
- Jumlah AllocatedUsageQuantity dari UsageAllocation harus sama denganUsageQuantity, yang merupakan penggunaan agregat.

Pengalaman pembeli

Tabel berikut menunjukkan contoh pengalaman pembeli setelah pembeli mengaktifkan tag `AccountId` dan `BusinessUnit` vendor.

Dalam contoh ini, pembeli dapat melihat penggunaan yang dialokasikan dalam Laporan Penggunaan Biaya mereka. Tag yang diukur vendor menggunakan awalan. “`aws:marketplace:isv`” Pembeli dapat mengaktifkannya di Billing and Cost Management, di bawah Tag Alokasi Biaya AWS, tag alokasi biaya yang dihasilkan.

Baris pertama dan terakhir dari Laporan Penggunaan Biaya relevan dengan apa yang dikirim Penjual ke Layanan Pengukuran (seperti yang ditunjukkan dalam [Pengalaman penjual](#) contoh).

Laporan Penggunaan Biaya (Sederhana)

ProductCode	Pembeli	UsageDimension	UsageQuantity	<code>aws:marketplace:isv:AccountId</code>	<code>aws:marketplace:isv:BusinessUnit</code>
xyz	111122223333	Jaringan: per (GB) diperiksa	70	2222	Operasi
xyz	111122223333	Jaringan: per (GB) diperiksa	30	3333	Keuangan
xyz	111122223333	Jaringan: per (GB) diperiksa	20	4444	IA
xyz	111122223333	Jaringan: per (GB) diperiksa	20	5555	Pemasaran
xyz	111122223333	Jaringan: per (GB) diperiksa	30	1111	Pemasaran

Untuk contoh kode, lihat [MeterUsagecontoh kode dengan penandaan alokasi penggunaan \(Opsional\)](#).

Contoh kode

Contoh kode berikut disediakan untuk membantu Anda mengintegrasikan produk kontainer Anda dengan yang AWS Marketplace APIs diperlukan untuk menerbitkan dan memelihara produk Anda.

MeterUsagecontoh kode dengan penandaan alokasi penggunaan (Opsional)

Contoh kode berikut relevan untuk produk kontainer dengan model harga konsumsi. Contoh Python mengirimkan catatan pengukuran dengan tag alokasi penggunaan yang sesuai untuk membebaskan biaya AWS Marketplace kepada pelanggan Anda. pay-as-you-go

```
# NOTE: Your application will need to aggregate usage for the
#       customer for the hour and set the quantity as seen below.
# AWS Marketplace can only accept records for up to an hour in the past.
#
# productCode is supplied after the AWS Marketplace Ops team has
# published the product to limited

# Import AWS Python SDK
import boto3
import time

usageRecord = [
    {
        "AllocatedUsageQuantity": 2,
        "Tags":
            [
                { "Key": "BusinessUnit", "Value": "IT" },
                { "Key": "AccountId", "Value": "123456789" },
            ]
    },
    {
        "AllocatedUsageQuantity": 1,
        "Tags":
            [
                { "Key": "BusinessUnit", "Value": "Finance" },
                { "Key": "AccountId", "Value": "987654321" },
            ]
    }
]
```

```
    }  
  ]  
  
  marketplaceClient = boto3.client("meteringmarketplace")  
  
  response = marketplaceClient.meter_usage(  
    ProductCode="testProduct",  
    Timestamp=int(time.time()),  
    UsageDimension="Dimension1",  
    UsageQuantity=3,  
    DryRun=False,  
    UsageAllocations=usageRecord  
  )
```

Untuk informasi selengkapnya `MeterUsage`, lihat [MeterUsage](#) di Referensi AWS Marketplace Metering Service API.

Contoh tanggapan

```
{ "MeteringRecordId": "string" }
```

Mengintegrasikan produk kontainer Anda menggunakan pengukuran khusus dengan dan AWS Marketplace Metering Service AWS SDK untuk Java

AWS Marketplace produk kontainer dapat memiliki pengukuran khusus hingga 24 dimensi harga yang berbeda per produk. Untuk mengaktifkan pengukuran khusus, Anda mengintegrasikan produk container Anda dengan AWS Marketplace Metering Service. Anda dapat menentukan unit harga Anda sendiri dan pengukuran kustom untuk penggunaan tersebut AWS untuk penagihan menggunakan operasi [MeterUsage](#) API. Contoh berikut menguraikan implementasi yang menggunakan AWS SDK untuk Java untuk mengintegrasikan dengan operasi [AWS Marketplace Metering Service](#) `MeterUsage`.

Untuk detail lengkap, lihat [MeterUsage Contoh Java](#). Banyak langkah-langkah berikut berlaku terlepas dari bahasa.

Contoh: Integrasi Layanan AWS Marketplace Metering

1. Masuk ke [Portal Manajemen AWS Marketplace](#).
2. Dari `Aset`, pilih `Kontainer` untuk mulai membuat produk kontainer baru. Membuat produk menghasilkan kode produk untuk produk untuk mengintegrasikan dengan citra kontainer Anda.

Untuk informasi tentang izin pengaturan AWS Identity and Access Management (IAM), lihat [the section called “AWS Marketplace izin API pengukuran dan hak”](#)

3. Unduh [AWS Java SDK](#) publik.

Important

Untuk memanggil operasi metering API dari Amazon Elastic Kubernetes Service (Amazon EKS), [Anda harus AWS menggunakan](#) SDK yang didukung dan berjalan di kluster Amazon EKS yang menjalankan Kubernetes 1.13 atau yang lebih baru.

4. Panggil `MeterUsage` operasi dari tugas atau pod setiap jam sekali untuk setiap penggunaan dimensi. Operasi API menerima satu catatan pengukuran untuk kombinasi `unikDimension`, `Resource`, dan `Hour`. Sumber daya adalah salah satu Amazon Elastic Container Service (Amazon ECS) tugas atau pod Amazon EKS.

```
{
  "ProductCode" : "string", // (required)
  "UsageDimension" : "string", // (required)
  "UsageQuantity": int, // (optional) Default is 0. Acceptable value from [0,
2147483647 (INT_MAX)]
  "Timestamp": Date, // (required) Timestamp in UTC. Value can be one hour in the
past.
  "UsageAllocations": List<UsageAllocation> // (optional) UsageAllocations across
1 or more tags.
}
```

Note

Dimungkinkan untuk melihat masalah sementara dalam menghubungkan ke AWS Marketplace Metering Service AWS Marketplace sangat merekomendasikan menerapkan percobaan ulang hingga 30 menit, dengan mundur eksponensial, untuk menghindari pemadaman jangka pendek atau masalah jaringan.

5. Buat kembali versi baru gambar kontainer Anda yang menyertakan `MeterUsage` panggilan, beri tag penampung, dan dorong ke registri Docker apa pun yang kompatibel dengan Amazon ECS atau Amazon EKS, seperti Amazon Elastic Container Registry (Amazon ECR). Jika Anda menggunakan Amazon ECR, pastikan bahwa akun meluncurkan tugas Amazon ECS atau Amazon EKS pod memiliki izin pada repositori Amazon ECR. Jika tidak, operasi gagal.

6. Buat [IAM](#) role yang memberikan izin untuk kontainer Anda untuk dipanggil `MeterUsage`, seperti yang didefinisikan dalam contoh kode berikut. Anda harus menyediakan peran ini AWS Identity and Access Management (IAM) dalam parameter [Peran Tugas tugas tugas](#) Amazon ECS atau definisi pod Amazon EKS.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "aws-marketplace:MeterUsage"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

7. Buat tugas Amazon ECS atau definisi pod Amazon EKS yang mereferensikan wadah yang telah terintegrasi AWS Marketplace dan mereferensikan peran IAM yang Anda buat di langkah 6. Jika Anda ingin melihat logging, aktifkan AWS CloudTrail logging dalam definisi tugas.
8. Buat Amazon ECS atau Amazon EKS cluster untuk menjalankan tugas Anda atau pod. Untuk informasi selengkapnya tentang membuat kluster Amazon ECS, lihat [Membuat kluster](#) di Panduan Pengembang Layanan Kontainer Elastis Amazon. Untuk informasi selengkapnya tentang membuat cluster Amazon EKS (menggunakan Kubernetes versi 1.1.3.x atau yang lebih baru), lihat [Membuat kluster EKS Amazon](#).
9. Konfigurasi cluster Amazon ECS atau Amazon EKS dan luncurkan definisi tugas Amazon ECS atau pod Amazon EKS yang Anda buat di langkah 8, di Wilayah us-east-1. AWS Hanya selama proses pengujian ini, sebelum produk ditayangkan, bahwa Anda harus menggunakan Wilayah ini.
10. Saat Anda mendapatkan respons yang valid dari `MeterUsage` untuk masing-masing dimensi yang diterbitkan untuk produk, Anda dapat mulai membuat produk kontainer Anda. Untuk pertanyaan, hubungi [AWS Marketplace Operasi Penjual](#) Tim.

MeterUsageContoh Java

Contoh kode berikut menggunakan AWS Marketplace Metering Service AWS SDK untuk Java dan AWS untuk memanggil MeterUsage operasi.

Contoh kode berikut panggilanMeterUsageoperasi tanpaUsageAllocations.

```
import com.amazonaws.services.marketplacemetering.AWSMarketplaceMetering;
import com.amazonaws.services.marketplacemetering.AWSMarketplaceMeteringClientBuilder;
import com.amazonaws.services.marketplacemetering.model.MeterUsageRequest;
import com.amazonaws.services.marketplacemetering.model.MeterUsageResult;

import java.util.Date;

public class MeterUsage {
    private static final String PRODUCT_CODE = ".....";
    private final AWSMarketplaceMetering awsMarketplaceMetering;

    public MeterUsage() {
        awsMarketplaceMetering =
AWSMarketplaceMeteringClientBuilder.standard().build();
    }

    /**
     * Submits metering record for a FCP Dimension. The API accepts 1 metering record
     per dimension
     * for a given buyer's resource for a given timestamp hour. Ex. If a buyer is
     running 10 tasks,
     * the API will accepts 1 call to MeterUsage in an hour for a given dimension for
     each running task.
     *
     * @param dimension - FCP dimension name provided during the publishing of the
     product.
     * @param quantity - FCP dimension consumption value for the hour.
     * @param timestamp - Timestamp, in UTC, for which the usage is being reported.
     * Timestamp cant be more than 1 hour in the past.
     * Make sure the timestamp value is not before the start of the
     software usage.
     */
    public void callMeterUsage(String dimension, int quantity, Date timestamp) {
        MeterUsageRequest meterUsageRequest = new MeterUsageRequest()
            .withProductCode(PRODUCT_CODE)
            .withUsageDimension(dimension)
            .withUsageQuantity(quantity)
    }
}
```

```
        .withTimestamp(timestamp);
    MeterUsageResult meterUsageResult =
awsMarketplaceMetering.meterUsage(meterUsageRequest);
    }
}
```

Contoh kode berikut panggilan `MeterUsageOperasi` dengan `UsageAllocations`.

```
private static String callMeterUsageWithAllocationsByTag(AWSMarketplaceMetering
marketplaceMetering) {
    // Tag Keys for the product
    String tagKey1 = "Key1";
    String tagKey2 = "Key2";
    String tagKey3 = "Key3";

    // 1st Usage Allocation bucket which has two Tags [{Key1, Key1Value1},{Key2,
Key2Value1}]
    List<Tag> tagsForUsageAllocation1 = Arrays.asList(new
Tag().withKey(tagKey1).withValue("Key1Value1"),
        new Tag().withKey(tagKey2).withValue("Key2Value1"));
    UsageAllocation usageAllocation1 = new UsageAllocation()
        .withTags(tagsForUsageAllocation1)
        .withAllocatedUsageQuantity(20);

    // 2nd Usage Allocation bucket which has two Tags [{Key1, Key1Value2},{Key2,
Key2Value1}]
    List<Tag> tagsForUsageAllocation2 = Arrays.asList(new
Tag().withKey(tagKey1).withValue("Key1Value2"),
        new Tag().withKey(tagKey2).withValue("Key2Value1"));
    UsageAllocation usageAllocation2 = new UsageAllocation()
        .withTags(tagsForUsageAllocation2)
        .withAllocatedUsageQuantity(20);

    // 3rd Usage Allocation bucket which has two Tags [{Key1, Key1Value2},{Key2,
Key2Value2},{Key3, Key3Value1}]
    List<Tag> tagsForUsageAllocation3 = Arrays.asList(new
Tag().withKey(tagKey1).withValue("Key1Value2"),
        new Tag().withKey(tagKey2).withValue("Key2Value2"),
        new Tag().withKey(tagKey3).withValue("Key3Value1"));
    UsageAllocation usageAllocation3 = new UsageAllocation()
        .withTags(tagsForUsageAllocation3)
        .withAllocatedUsageQuantity(15);
}
```

```
// 4th Usage Allocation bucket with no tags
UsageAllocation usageAllocation4 = new UsageAllocation()
    .withAllocatedUsageQuantity(15);

List<UsageAllocation> usageAllocationList = Arrays.asList(usageAllocation1,
    usageAllocation2,
    usageAllocation3,
    usageAllocation4);

MeterUsageRequest meterUsageRequest = new MeterUsageRequest()
    .withProductCode("TestProductCode")
    .withUsageDimension("Dimension1")
    .withTimestamp(new Date())
    //UsageQuantity value must match with sum of all AllocatedUsageQuantity
    .withUsageQuantity(70)
    .withUsageAllocations(usageAllocationList);

MeterUsageResult meterUsageResult;
try {
    meterUsageResult = marketplaceMetering.meterUsage(meterUsageRequest);
} catch (Exception e) {
    // Log Error
    throw e;
}

return meterUsageResult.getMeteringRecordId();
}
```

Harga kontrak untuk produk kontainer dengan AWS License Manager

Untuk produk berbasis kontainer dengan harga kontrak, Anda dapat menggunakan AWS License Manager untuk mengaitkan lisensi dengan produk Anda. AWS License Manager adalah alat manajemen lisensi yang memungkinkan aplikasi Anda untuk melacak dan memperbarui lisensi (juga dikenal sebagai hak) yang telah dibeli oleh pelanggan. Bagian ini memberikan informasi tentang cara mengintegrasikan produk Anda AWS License Manager. Setelah integrasi selesai, Anda dapat mempublikasikan daftar produk Anda AWS Marketplace.

Jika Anda mengintegrasikan License Manager dengan produk AWS Marketplace for Containers Anywhere untuk Amazon EKS Anywhere, Amazon ECS Anywhere, Amazon Elastic Compute Cloud (Amazon EC2), atau infrastruktur lokal, ikuti petunjuknya. [Mengintegrasikan AWS Marketplace untuk Kontainer Di Mana Saja dengan License Manager](#)

Untuk informasi selengkapnya AWS License Manager, lihat [Panduan AWS License Manager Pengguna](#) dan [AWS License Manager](#) bagian Referensi AWS CLI Perintah.

Untuk informasi selengkapnya tentang integrasi AWS License Manager dengan produk kontainer dengan harga kontrak, lihat lab [Integrasikan dengan pembayaran di muka](#) bengkel AWS Marketplace penjual.

Topik

- [Model lisensi](#)
- [AWS License Manager prasyarat integrasi](#)
- [Mengintegrasikan produk kontainer dengan License Manager](#)
- [Operasi API License Manager](#)
- [Perpanjangan dan peningkatan lisensi](#)
- [Mengintegrasikan AWS Marketplace untuk Kontainer Di Mana Saja dengan License Manager](#)

Model lisensi

AWS Marketplace integrasi dengan AWS License Manager mendukung dua model lisensi:

- [Model lisensi yang dapat dikonfigurasi](#)
- [Model lisensi berjenjang](#)

Model lisensi yang dapat dikonfigurasi

Model lisensi yang dapat dikonfigurasi (juga dikenal sebagai model lisensi yang dapat diukur) memberikan hak kepada pembeli untuk sejumlah sumber daya tertentu setelah pembeli memperoleh lisensi.

Anda menetapkan dimensi harga dan harga per unit. Kemudian, pembeli dapat memilih jumlah sumber daya yang ingin mereka beli.

Example dimensi harga dan harga per unit

Anda dapat menetapkan dimensi harga (seperti cadangan data) dan harga per unit (seperti \$30/unit).

Pembeli dapat memilih untuk membeli 5, 10, atau 20 unit.

Produk Anda melacak dan penggunaan meter untuk mengukur jumlah sumber daya yang dikonsumsi.

Dengan model konfigurasi, hak dihitung dalam salah satu dari dua cara:

- [Lisensi drawdown](#)
- [Lisensi mengambang](#)

Lisensi drawdown

Lisensi diambil dari kumpulan jumlah lisensi yang diizinkan saat digunakan. Hak tersebut diperiksa secara permanen dan tidak dapat dikembalikan ke kumpulan lisensi.

Example Memproses data dalam jumlah terbatas

Seorang pengguna berhak untuk memproses 500 GB data. Ketika mereka terus memproses data, kuantitas diambil dari kumpulan 500 GB hingga semua lisensi 500 GB dikonsumsi.

Untuk lisensi drawdown, Anda dapat menggunakan operasi CheckoutLicense API untuk memeriksa unit lisensi (hak) yang digunakan.

Example cadangan ke Amazon S3 untuk sejumlah unit/tahun

Anda memiliki produk penyimpanan yang memungkinkan pencadangan ke Amazon Simple Storage Service hingga 1.024 unit untuk data selama satu tahun. Aplikasi Anda dapat diluncurkan dengan menggunakan beberapa EC2 instans Amazon. Aplikasi Anda memiliki mekanisme untuk melacak dan mengumpulkan data. Perangkat lunak Anda memanggil operasi CheckoutLicense API dengan ID Produk pada setiap cadangan atau pada interval tetap untuk memperbarui jumlah yang dikonsumsi.

Dalam contoh ini, perangkat lunak Anda memanggil operasi CheckoutLicense API untuk memeriksa 10 unit data. Ketika total kapasitas mencapai batas cadangan yang telah dibeli pelanggan, panggilan API gagal.

Permintaan

```
linux-machine ~]$ aws license-manager checkout-license\  
--product-sku "2205b290-19e6-4c76-9eea-377d6bf71a47" \  
--checkout-type "PERPETUAL" \  
--key-fingerprint "aws:294406891311:AWS/Marketplace:issuer-fingerprint" \  

```

```
--entitlements "Name=DataConsumption, Value=10, Unit=Count" \  
--client-token "AKIAIOSFODNN7EXAMPLE"
```

Respons

```
{"CheckoutType": "PERPETUAL",  
"EntitlementsAllowed": [{  
  "Name": "IntermediateTier",  
  "Units": "None"  
}],  
"Expiration": "2021-04-22T19:02:36",  
"IssuedAt": "2021-04-22T18:02:36",  
"LicenseArn": "arn:aws:license-manager::294406891311:license:l-16bf01b...",  
"LicenseConsumptionToken": "AKIAIOSFODNN7EXAMPLE"  
}
```

Lisensi mengambang

Lisensi dikembalikan ke kumpulan jumlah lisensi yang diizinkan setelah digunakan.

Untuk lisensi mengambang, aplikasi memeriksa hak dari kumpulan hak menggunakan operasi `CheckoutLicense` API saat sumber daya digunakan. Respons operasi `CheckoutLicense` API mencakup token konsumsi lisensi yang merupakan pengenal unik untuk checkout. Token konsumsi lisensi dapat digunakan untuk melakukan tindakan tambahan pada hak yang diperiksa, seperti memeriksanya kembali ke lisensi atau memperpanjang checkout.

Untuk memeriksa kembali hak ke dalam kumpulan, gunakan operasi `CheckInLicense` API saat sumber daya tidak lagi digunakan.

```
aws license-manager check-in-license --license-consumption-token  
"f1603b3c1f574b7284db84..."
```

Jika terjadi kegagalan untuk memeriksa hak (jika aplikasi macet), hak akan memeriksa kembali ke kumpulan secara otomatis setelah 60 menit. Jika sumber daya digunakan lebih dari 60 menit, itu adalah praktik terbaik untuk menjaga hak diperiksa dari kumpulan dengan menggunakan operasi `ExtendLicenseConsumption` API selama sumber daya digunakan.

```
aws license-manager extend-license-consumption --license-consumption-token  
"f1603b3c1f574b7284..."
```

Example jumlah pengguna dari batas atas tetap

Seorang pengguna berhak atas 500 pengguna simultan pada aplikasi. Saat pengguna masuk dan keluar, pengguna ditarik dan dikembalikan ke kumpulan 500 pengguna. Namun, aplikasi tidak dapat menarik lebih dari 500 pengguna dari pool karena 500 pengguna simultan adalah batas atas tetap.

Untuk hak mengambang, Anda dapat menggunakan operasi `CheckInLicense` API untuk mengembalikan unit lisensi ke kumpulan hak.

Example jumlah pengguna bersamaan selama satu tahun

Harga produk Anda berdasarkan jumlah pengguna bersamaan. Pelanggan membeli lisensi untuk 10 pengguna selama satu tahun. Pelanggan meluncurkan perangkat lunak dengan memberikan izin AWS Identity and Access Management (IAM). Ketika pengguna log in, aplikasi Anda memanggil operasi `CheckoutLicense` API untuk mengurangi kuantitas sebesar 1. Saat pengguna log out, aplikasi mengembalikan lisensi tersebut ke pool dengan memanggil operasi `CheckInLicense` API. Jika Anda tidak menelepon `CheckInLicense`, unit lisensi akan secara otomatis diperiksa setelah 1 jam.

Note

Dalam Permintaan berikut, `key-fingerprint` bukan nilai placeholder tetapi nilai sebenarnya dari sidik jari yang dengannya semua lisensi akan dipublikasikan.

Permintaan

```
aws license-manager checkout-license\  
--product-sku "2205b290-19e6-4c76-9eea-377d6bf71a47" \  
--checkout-type "PROVISIONAL" \  
--key-fingerprint "aws:294406891311:AWS/Marketplace:issuer-fingerprint" \  
--entitlements "Name=ReadOnlyUSers, Value=10, Unit=Count" \  
--client-token "AKIAIOSFODNN7EXAMPLE"
```

Respons

```
{  
  "CheckoutType": "PROVISIONAL",  
  "EntitlementsAllowed": [  
    {  
      "Name": "ReadOnlyUsers",
```

```

    "Count": 10,
    "Units": "Count",
    "Value": "Enabled"
  }
},
"Expiration": "2021-04-22T19:02:36",
"IssuedAt": "2021-04-22T18:02:36",
"LicenseArn": "arn:aws:license-manager::294406891311:license:l-16bf01b...",
"LicenseConsumptionToken": "AKIAIOSFODNN7EXAMPLE"
}

```

Model lisensi berjenjang

Model lisensi berjenjang memberikan hak kepada pembeli ke tingkat tertentu, atau tingkat, fitur aplikasi setelah pembeli memperoleh lisensi.

Anda membuat tingkatan untuk produk Anda, seperti Basic, Intermediate, dan Premium. Pembeli kemudian memilih salah satu tingkatan yang telah ditentukan.

Aplikasi tidak perlu melacak atau mengukur penggunaan aplikasi.

Dengan model lisensi berjenjang, hak tidak dihitung melainkan menandakan tingkat layanan yang diperoleh oleh pelanggan.

Jika Anda ingin menawarkan fitur yang dibundel bersama-sama, tingkatan lebih disukai.

Example Tingkat Dasar, Menengah, dan Premium

Pelanggan dapat menandatangani kontrak untuk salah satu dari tiga kemungkinan tingkatan perangkat lunak: Dasar, Menengah, atau Premium. Masing-masing tingkatan ini memiliki harga sendiri. Perangkat lunak Anda dapat mengidentifikasi tingkat yang telah didaftarkan pelanggan dengan menjalankan operasi CheckoutLicense API dan menentukan semua tingkatan yang mungkin dalam permintaan.

Tanggapan permintaan berisi hak yang sesuai dengan tingkat yang telah diperoleh pelanggan. Berdasarkan informasi ini, perangkat lunak dapat memberikan pengalaman pelanggan yang sesuai.

Permintaan

```

linux-machine ~]$ aws license-manager checkout-license \
--product-sku "2205b290-19e6-4c76-9eea-377d6bf71a47" \
--checkout-type "PROVISIONAL" \
--key-fingerprint "aws:294406891311:AWS/Marketplace:issuer-fingerprint" \

```

```
--entitlements "Name=BasicTier, Unit=None" "Name=IntermediateTier, Unit=None" \
"Name=PremiumTier, Unit=None"
```

Respons

```
{
  "CheckoutType": "PROVISIONAL",
  "EntitlementsAllowed": [
    {
      "Name": "IntermediateTier",
      "Units": "None"
    }
  ],
  "Expiration": "2021-04-22T19:02:36",
  "IssuedAt": "2021-04-22T18:02:36",
  "LicenseArn": "arn:aws:license-manager::294406891311:license:l-16bf01b...",
  "LicenseConsumptionToken": "AKIAIOSFODNN7EXAMPLE"
}
```

AWS License Manager prasyarat integrasi

Sebelum menerbitkan produk, Anda harus melakukan hal berikut:

1. Buat produk kontainer baru di Portal Manajemen AWS Marketplace, dan catat kode produknya.

Untuk informasi selengkapnya, lihat [Ikhtisar: Buat produk kontainer](#).

2. Gunakan peran IAM untuk tugas atau pod yang menjalankan aplikasi Anda dengan izin IAM yang diperlukan untuk memanggil operasi `CheckoutLicense`, `ExtendLicenseConsumption`, dan `CheckInLicense` API.

Izin IAM yang diperlukan dirinci dalam kebijakan IAM berikut.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "VisualEditor0",
      "Effect": "Allow",
      "Action": [
```

```

        "license-manager:CheckoutLicense",
        "license-manager:GetLicense",
        "license-manager:CheckInLicense",
        "license-manager:ExtendLicenseConsumption",
        "license-manager:ListReceivedLicenses"
    ],
    "Resource": "*"
}
]
}

```

3. Lakukan panggilan uji ke operasi RegisterUsage API dengan catatan untuk semua dimensi harga yang Anda tentukan.

Mengintegrasikan produk kontainer dengan License Manager

Untuk mengintegrasikan produk berbasis kontainer Anda dengan License Manager

1. Setel izin IAM untuk memanggil License Manager. Untuk informasi selengkapnya, lihat [AWS License Manager prasyarat integrasi](#).
2. Unduh AWS SDK.

Note

Jangan mengkonfigurasi AWS kredensial dalam perangkat lunak Anda. AWS kredensial untuk pembeli diperoleh secara otomatis saat runtime saat container Anda berjalan dalam instans Amazon, tugas Amazon ECS EC2, atau pod Amazon EKS.

3. Tambahkan cek lisensi ke produk Anda.

Produk Anda dapat memanggil operasi CheckoutLicense API di mana pun pemeriksaan lisensi harus dilakukan. Untuk memeriksa lisensi, produk Anda harus tahu:

1. Penerbit lisensi tepercaya (AWS Marketplace)
2. SKU Produk aplikasi (ID Produk)
3. Hak untuk memeriksa aplikasi ini

Panggilan API bervariasi berdasarkan jenis lisensi harga yang Anda siapkan.

4. Publikasikan daftar produk Anda di AWS Marketplace.

Operasi API License Manager

Untuk mengelola lisensi yang disimpan di akun License Manager pelanggan, perangkat lunak Anda dapat menggunakan operasi API berikut:

- **GetLicense**— API yang dapat ditanyakan oleh perangkat lunak. Ini mengambil status lisensi yang dibeli (yaitu kedaluwarsa atau segera kedaluwarsa) dan mengirimkan pemberitahuan status kepada pelanggan.
- **CheckoutLicense**— Menemukan lisensi yang telah dibeli pengguna. Anda juga dapat menggunakan operasi **CheckoutLicense** API untuk memperbarui kuantitas lisensi ketika pengguna telah menggunakan sejumlah lisensi. Dengan **CheckoutLicense**, Anda dapat terus memeriksa jumlah lisensi yang digunakan oleh pelanggan. Ketika pelanggan kehabisan semua lisensi, panggilan ini mengembalikan kesalahan. Untuk informasi tentang irama yang disarankan untuk dijalankan **CheckoutLicense**, lihat [the section called “Perpanjangan dan peningkatan lisensi”](#).
- **ExtendLicenseConsumption**— Dalam hal dimensi mengambang, ketika perangkat lunak memeriksa lisensi, lisensi akan kembali ke kolam secara otomatis setelah 60 menit. Jika Anda ingin memperpanjang waktu lisensi tetap diperiksa, gunakan operasi **ExtendLicenseConsumption** API untuk memperpanjang lisensi selama 60 menit lagi.
- **CheckInLicense**— Dalam kasus dimensi mengambang, ketika Anda ingin mengembalikan lisensi ke kumpulan hak, gunakan operasi **CheckInLicense** API.
- **ListReceivedLicensesAPI** - Daftar lisensi yang dibeli oleh pembeli.

Perpanjangan dan peningkatan lisensi

Pelanggan dapat memperbarui atau meningkatkan lisensi mereka di Portal Manajemen AWS Marketplace Setelah mereka melakukan pembelian tambahan AWS Marketplace , buat versi baru dari lisensi yang mencerminkan hak baru. Perangkat lunak Anda membaca hak baru dengan menggunakan operasi API yang sama. Anda tidak perlu melakukan sesuatu yang berbeda dalam hal integrasi License Manager untuk menangani pembaruan dan peningkatan.

Karena perpanjangan lisensi, peningkatan, pembatalan, dan sebagainya, kami menyarankan agar produk Anda memanggil operasi **CheckoutLicense** API dengan irama reguler saat produk sedang digunakan. Dengan menggunakan operasi **CheckoutLicense** API pada irama reguler, produk dapat mendeteksi perubahan hak seperti upgrade dan kedaluwarsa.

Kami menyarankan Anda melakukan panggilan **CheckoutLicense** API setiap 15 menit.

Mengintegrasikan AWS Marketplace untuk Kontainer Di Mana Saja dengan License Manager

Sebagai AWS Marketplace penjual, Anda dapat berintegrasi AWS License Manager dengan produk AWS Marketplace for Containers Anywhere untuk Amazon EKS Anywhere, Amazon ECS Anywhere, EC2 Amazon, atau infrastruktur lokal. Bagian berikut memberikan instruksi untuk integrasi ini.

Untuk informasi umum tentang integrasi License Manager dengan AWS Marketplace, termasuk model lisensi yang tersedia, lihat [Harga kontrak untuk produk kontainer dengan AWS License Manager](#). Untuk informasi selengkapnya AWS License Manager, lihat [Panduan AWS License Manager Pengguna](#) dan [AWS License Manager](#) bagian Referensi AWS CLI Perintah.

Topik

- [Mengintegrasikan produk AWS Marketplace for Containers Anywhere dengan License Manager](#)
- [Menguji integrasi License Manager secara lokal](#)
- [Menguji integrasi License Manager di Amazon EKS](#)
- [Hak lisensi mengambang dengan License Manager](#)
- [Praktik terbaik untuk mengintegrasikan dengan License Manager untuk penerapan lokal](#)
- [LicenseManagerCredentialsProvider- Implementasi Java](#)
- [LicenseManagerCredentialsProvider- Golang implementasi](#)

Mengintegrasikan produk AWS Marketplace for Containers Anywhere dengan License Manager

Gunakan petunjuk berikut untuk mengintegrasikan produk AWS Marketplace for Containers Anywhere Anda AWS License Manager.

Untuk mengintegrasikan produk Containers Anywhere Anda AWS Marketplace dengan License Manager

1. Buka browser web dan masuk ke file [Portal Manajemen AWS Marketplace](#).
2. Buat ID produk untuk produk kontainer Anda dengan melakukan langkah-langkah berikut. Anda akan menggunakan ID ini dalam gambar kontainer Anda untuk pemeriksaan lisensi di langkah selanjutnya.
 - a. Dari bilah menu, perluas Aset, dan pilih Kontainer.
 - b. Masukkan nama yang menghadap pelanggan untuk produk Anda, dan pilih Buat. Anda dapat mengubah nama ini nanti.

- c. Catat ID Produk. Anda akan menggunakannya saat membuat atau memperbarui detail harga produk.

 Tip

Jika Anda kehilangan ID produk Anda, Anda dapat menemukannya di Portal Manajemen AWS Marketplace dengan memilih Container dari menu Assets. Halaman Kontainer menampilkan daftar produk Anda dengan produk terkait mereka IDs.

3. Unduh AWS SDK publik terbaru dan kemudian instal di aplikasi penampung Anda. Anda dapat menemukan petunjuk penginstalan untuk AWS SDK pilihan Anda di [Tools to Build on AWS](#).

 Note

Untuk memanggil operasi License Manager API dari Amazon EKS Anywhere atau kluster Kubernetes yang tidak disediakan oleh AWS, Anda harus menggunakan SDK yang didukung. AWS Untuk melihat daftar yang didukung AWS SDKs, lihat [Menggunakan AWS SDK yang didukung](#).

4. Buat AWS License Manager klien dengan penyedia kredensi khusus sehingga dapat memberikan kredensial ke aplikasi kontainer yang digunakan di AWS maupun lokal. Untuk kode sumber lengkap untuk penyedia kredensi kustom `LicenseCredentialProvider`, lihat bagian berikut:

- [LicenseManagerCredentialsProvider- Implementasi Java](#)
- [LicenseManagerCredentialsProvider- Golang implementasi](#)

`LicenseCredentialProvider` memperluas rantai penyedia kredensi default AWS SDK untuk penggunaan lokal dengan menambahkan.

`LicenseManagerTokenCredentialsProvider` ini memberikan kredensial dengan menggunakan License Manager OIDC mengeluarkan token identitas di lingkungan lokal. Anda harus menyertakan kode sumber untuk `LicenseCredentialProvider` di classpath aplikasi Anda.

Note

Memperluas aplikasi kontainer `DefaultCredentialsProvider` memungkinkan aplikasi kontainer yang sama untuk mendapatkan kredensial saat berjalan AWS dan saat berjalan di lingkungan lokal. Jika aplikasi kontainer sudah menggunakan rantai penyedia kredensi kustom alih-alih default, itu juga dapat diperpanjang dengan menambahkan `LicenseManagerTokenCredentialsProvider` ke rantai kustom.

Cuplikan kode berikut adalah contoh membuat AWS License Manager klien menggunakan Java.

```
LicenseManagerClientBuilder clientBuilder =  
    LicenseManagerClient.builder().credentialsProvider(LicenseCredentialsProvider.create());
```

5. Panggil operasi `CheckoutLicense` API dengan menggunakan `aws license-manager checkout-license` perintah dari setiap image container berbayar dalam penawaran produk Anda. Ini memeriksa bahwa pembeli berhak menggunakan lisensi untuk aplikasi Anda. Jika pembeli berhak atas aplikasi, `CheckoutLicense` berhasil dan mengembalikan hak yang diminta dan nilainya. Jika pembeli tidak berhak atas aplikasi, `CheckoutLicense` melempar pengecualian.

Parameter berikut diperlukan saat memanggil operasi `CheckoutLicense` API:

- `CheckoutType`— Nilai yang valid adalah `PROVISIONAL` atau `PERPETUAL`:
 - Gunakan `PERPETUAL` ketika jumlah hak yang diperiksa akan habis dari kolam.

Contoh: Pembeli berhak memproses 500 GB data. Ketika mereka terus memproses data, kuantitas ditarik ke bawah dan habis dari kumpulan 500 GB.

- Gunakan `PROVISIONAL` untuk hak lisensi mengambang di mana hak diperiksa keluar dari kumpulan dan dikembalikan setelah digunakan.

Contoh: Pengguna berhak atas 500 pengguna simultan pada aplikasi. Saat pengguna masuk atau keluar, pengguna ditarik atau dikembalikan ke kumpulan 500 pengguna. Untuk mempelajari lebih lanjut tentang hak lisensi mengambang, lihat [Hak lisensi mengambang dengan License Manager](#)

- `ClientToken`— Pengidentifikasi unik dan peka huruf besar/kecil. Sebaiknya gunakan UUID acak untuk setiap permintaan unik.

- **Entitlements**— Daftar hak yang akan diperiksa.
- Untuk hak fitur, berikan Name dan Unit properti sebagai berikut.

```
{
  "Name": "<Entitlement_Name>",
  "Unit": "None"
}
```

- Untuk hak yang dihitung, berikan, NameUnit, dan Count properti sebagai berikut.

```
{
  "Name": "<Entitlement_Name>",
  "Unit": "<Entitlement_Unit>",
  "Value": <Desired_Count>
}
```

- **KeyFingerprint**— Sidik jari kunci untuk lisensi yang dikeluarkan oleh AWS Marketplace adalah `aws:294406891311:AWS/Marketplace:issuer-fingerprint`. Menggunakan sidik jari kunci ini memastikan bahwa lisensi dikeluarkan oleh AWS Marketplace dan bukan oleh entitas yang tidak dapat diandalkan.
- **ProductSKU**— ID Produk yang dihasilkan Portal Manajemen AWS Marketplace pada langkah sebelumnya.

Cuplikan berikut adalah contoh panggilan menggunakan operasi `CheckoutLicense` API menggunakan AWS CLI

```
aws license-manager checkout-license \
--product-sku "2205b290-19e6-4c76-9eea-377d6bf71a47" \
--checkout-type "PROVISIONAL" \
--client-token "79464194dca9429698cc774587a603a1" \
--entitlements "Name=AWS::Marketplace::Usage/Drawdown/DataConsumption, Value=10,
Unit=Gigabytes" \
--key-fingerprint "aws:294406891311:AWS/Marketplace:issuer-fingerprint"
```

Note

Untuk memeriksa lisensi, aplikasi kontainer memerlukan akses jaringan keluar untuk menggunakan License Manager. Aplikasi yang digunakan di lokasi mungkin mengalami akses jaringan keluar yang tidak dapat diandalkan atau lambat. Aplikasi ini harus

menyertakan percobaan ulang yang memadai saat memanggil License Manager. Untuk informasi selengkapnya, lihat [Praktik terbaik untuk mengintegrasikan dengan License Manager untuk penerapan lokal](#).

6. Hubungi operasi CheckoutLicense API dengan irama reguler untuk mengidentifikasi perubahan apa pun pada lisensi pelanggan karena perpanjangan, peningkatan, atau pembatalan yang dilakukan. AWS Marketplace Irama tergantung pada aplikasi. Kami merekomendasikan untuk memeriksa lisensi sekali sehari untuk mengambil perubahan secara otomatis tanpa campur tangan pembeli.

Aplikasi yang digunakan di lokasi mungkin memiliki akses jaringan keluar yang tidak dapat diandalkan untuk memeriksa lisensi pada irama reguler. Dalam kasus seperti itu, aplikasi harus menggunakan lisensi cache untuk ketahanan yang memadai. Untuk informasi selengkapnya, lihat [Praktik terbaik untuk mengintegrasikan dengan License Manager untuk penerapan lokal](#).

7. Setelah Anda mengintegrasikan CheckoutLicense panggilan dengan aplikasi container Anda, buat versi baru image container Docker Anda dengan perubahan.
8. Perbarui bagan Helm aplikasi Anda untuk menerima rahasia Kubernetes sebagai input opsional yang berisi konfigurasi untuk mengakses lisensi menggunakan License Manager. APIs Rahasia konfigurasi akan berisi token identitas yang dikeluarkan oleh License Manager dan AWS Identity and Access Management peran yang akan digunakan oleh penyedia kredensi kustom yang dijelaskan sebelumnya untuk mendapatkan AWS kredensial untuk memanggil License Manager APIs saat aplikasi kontainer di-deploy di lokasi. Juga, tambahkan Wilayah AWS sebagai input dengan nilai defaultus-east-1.

Pembeli yang menerapkan aplikasi kontainer di tempat dapat membuat rahasia Kubernetes melalui pengalaman AWS Marketplace pembeli untuk produk kontainer. Berikan nama rahasia Kubernetes sebagai input ke perintah. `helm install` Rahasia konfigurasi dikonfigurasi dalam format berikut.

```
apiVersion: v1
kind: Secret
metadata:
  name: aws-marketplace-license-config
type: Opaque
stringData:
  license_token: <token_value> // License Manager issued JWT token
  iam_role: <role_arn> // AWS Identity and Access Management role to assume with
  license token
```

9. Perbarui template penerapan aplikasi di bagan Helm untuk gambar kontainer yang terintegrasi dengan menyertakan AWS License Manager yang berikut:

- Akun layanan untuk pod - Akun layanan diperlukan untuk penerapan Helm di Amazon EKS. Ini digunakan untuk mendapatkan izin untuk memanggil operasi License Manager API dengan menyiapkan peran IAM untuk akun layanan pada image container. Untuk informasi selengkapnya tentang peran IAM untuk akun layanan, lihat [peran IAM untuk akun layanan](#).
- Akses lisensi untuk penerapan lokal — Rahasia konfigurasi lisensi diperlukan untuk menyediakan kredensial dan izin yang sesuai untuk memanggil operasi License Manager API untuk penerapan Helm di lingkungan lokal. Pembeli akan menghasilkan dan memberikan rahasia lisensi kepada Helm dari pengalaman AWS Marketplace pembeli.

Cuplikan kode berikut adalah spesifikasi penyebaran sampel dengan akun layanan, konfigurasi lisensi, dan rahasia tarik gambar.

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: example-app
spec:
  replicas: 1
  selector:
    matchLabels:
      app: example-app
  template:
    metadata:
      labels:
        app: example-app
spec:
  // Service account for pod
  serviceAccountName: {{ .Values.serviceAccountName }}
  containers:
    - name: example-app
      image: example-app
      ports:
        - containerPort: 8001
  // Add the following conditional attributes
  {{ - if .Values.awsmpl.licenseConfigSecretName }}
    //Mount the license volume to the container image
    volumeMounts:
      - name: awsmpl-product-license
```

```

        mountPath: "/var/run/secrets/product-license"
    //Add following environment variable to container for credential
provider
    env:
      - name: AWS_WEB_IDENTITY_REFRESH_TOKEN_FILE
        value: "/var/run/secrets/product-license/license_token"
      - name: AWS_ROLE_ARN
        valueFrom:
          secretKeyRef:
            name: {{ .Values.aws.licenseConfigSecretName }}
            key: iam_role
    //Mount the license secret as a volume to the pod
    volumes:
      - name: awsmc-product-license
        secret:
          secretName: {{ .Values.aws.licenseConfigSecretName }}
          optional: true
    {{ - end }}

```

Note

Rahasia konfigurasi lisensi adalah opsional. Pembeli hanya menggunakan nilai untuk penerapan lokal. Untuk AWS penerapan, spesifikasi penyebaran harus menyertakan akun layanan untuk gambar terintegrasi License Manager.

10. Uji integrasi License Manager secara lokal dan di Amazon EKS dengan melakukan langkah-langkah di bagian berikut:
 - a. [Menguji integrasi License Manager secara lokal](#)
 - b. [Menguji integrasi License Manager di Amazon EKS](#)
11. Setelah berhasil memverifikasi integrasi License Manager baik di dalam AWS maupun lokal, Anda dapat membuat daftar produk container dengan mengikuti langkah-langkah di dalamnya [khtisar: Buat produk kontainer](#).

Menguji integrasi License Manager secara lokal

Anda dapat menggunakan minikube atau penyiapan lainnya untuk menguji integrasi License Manager pada kluster Kubernetes mana pun secara lokal. Pastikan bahwa kluster Kubernetes memiliki akses internet keluar untuk memanggil operasi License Manager API.

Untuk menguji integrasi License Manager secara lokal

1. Buat lisensi pengujian di akun penjual uji dengan hak yang diinginkan. Untuk menyiapkan lisensi pengujian, lihat [CreateLicense](#) di Referensi AWS License Manager API. Atau, gunakan skrip berikut untuk membuat lisensi pengujian dan kemudian buat hibah lisensi ke akun pembeli uji untuk menggunakan lisensi. Skrip berikut menggunakan kredensial akun penjual uji.

```

read -p 'AWS Account for test buyer: ' TEST_BUYER_ACCOUNT_ID
read -p 'License entitlements: ' ENTITLEMENTS

# TEST_SELLER_ACCOUNT_ID="109876543210"
# ENTITLEMENTS="{\"Name\": \"ByData\", \"MaxCount\": 1000, \"Overage\": true, \"Unit\": \"Gigabits\", \"AllowCheckIn\": true}"

# Create License

NOW=$(date +%Y-%m-%dT00:00:00+00:00)

PRODUCT_NAME="My awesome product"
PRODUCT_SKU="c97b7825-44c4-4f42-b025-12baa4c171e0"

LICENSE_BENEFICIARY=" arn:aws:iam::$TEST_BUYER_ACCOUNT_ID:root "
LICENSE_ISSUER_NAME="test-seller"
LICENSE_NAME="test-seller-license"

CLIENT_TOKEN="b3920968-a94f-4547-af07-3dd232319367"
CONSUMPTION_TTL=180
CONSUMPTION_RENEW_TYPE="None"

HOME_REGION="us-east-1"

LICENSE_ARN=$(aws license-manager create-license --license-name
"$LICENSE_NAME" --product-name "$PRODUCT_NAME" --product-sku
"$PRODUCT_SKU" --issuer Name="$LICENSE_ISSUER_NAME" --home-region
"$HOME_REGION" --validity Begin="$NOW" --entitlements "$ENTITLEMENTS"
--beneficiary "$LICENSE_BENEFICIARY" --consumption-configuration
RenewType="$CONSUMPTION_RENEW_TYPE",ProvisionalConfiguration={MaxTimeToLiveInMinutes=$CONSUMPTION_TTL} --client-token "$CLIENT_TOKEN" | jq -r ".LicenseArn" )

echo "License arn: $LICENSE_ARN"

# Create Grant

```

```

GRANT_TOKEN="e9a14140-4fca-4219-8230-57511a6ea6"
GRANT_NAME="test-grant"

GRANT_ARN=$(aws license-manager create-grant --grant-name "$GRANT_NAME"
  --license-arn "$LICENSE_ARN" --principals "$LICENSE_BENEFICIARY" --home-
  region "$HOME_REGION" --client-token "$GRANT_TOKEN" --allowed-operations
  "CheckoutLicense" "CheckInLicense" "ExtendConsumptionLicense" "CreateToken" | jq -
  r ".GrantArn")

echo "Grant arn: $GRANT_ARN"

```

2. Buat rahasia Kubernetes dengan token lisensi dan peran IAM menggunakan format rahasia yang ditentukan sebelumnya. Gunakan operasi License Manager CreateToken API untuk menghasilkan token lisensi. Kemudian, gunakan operasi IAM CreateRole API untuk membuat peran IAM dengan izin dan kebijakan kepercayaan. Lihat contoh dalam skrip berikut. Skrip berikut menggunakan kredensial akun pembeli uji.

```

read -p 'AWS Account for test license: ' TEST_ACCOUNT_ID
read -p 'License Arn' LICENSE_ARN
# Create IAM Role
ROLE_NAME="AWSLicenseManagerConsumptionTestRole"
ROLE_DESCRIPTION="Role to test AWS License Manager integration on-prem"
ROLE_POLICY_ARN="arn:aws:iam::aws:policy/service-role/
AWSLicenseManagerConsumptionPolicy"
ROLE_TRUST_POLICY="{\"Version\": \"2012-10-17\", \"Statement\": [{ \"Effect\":
  \"Allow\", \"Principal\": { \"Federated\": \"openid-license-manager.amazonaws.com
  \" }, \"Action\": \"sts:AssumeRoleWithWebIdentity\", \"Condition\":
  { \"ForAnyValue:StringLike\": { \"openid-license-manager.amazonaws.com:amr\":
  \"aws:license-manager:token-issuer-account-id:${TEST_ACCOUNT_ID}\" } } ] }"
ROLE_SESSION_DURATION=3600

ROLE_ARN=$(aws iam create-role --role-name "$ROLE_NAME" --description
  "$ROLE_DESCRIPTION" --assume-role-policy-document "$ROLE_TRUST_POLICY" --max-
  session-duration $ROLE_SESSION_DURATION | jq ".Role" | jq -r ".Arn")

aws iam attach-role-policy --role-name "$ROLE_NAME" --policy-arn "$ROLE_POLICY_ARN"

echo "Role arn: $ROLE_ARN"

# Create Token
CLIENT_TOKEN="b3920968-a94f-4547-af07-3dd232319367"

```

```
TOKEN=$(aws license-manager create-token --license-arn $LICENSE_ARN --role-arns
  $ROLE_ARN --client-token $CLIENT_TOKEN | jq '.Token')

echo "License access token: $TOKEN"
```

3. Siapkan kluster Kubernetes yang dihosting di luar AWS Gunakan untuk menguji bahwa aplikasi kontainer dapat terhubung ke AWS License Manager API dari lingkungan selain AWS dan bahwa penyedia kredensi khusus terintegrasi dengan baik dalam aplikasi.
4. Menerapkan token lisensi dan peran IAM yang dihasilkan sebelumnya ke dalam kluster Kubernetes lokal.

```
kubectl create secret generic "awsmp-license-access-config" \
  --from-literal=license_token=${TOKEN} \
  --from-literal=iam_role=${ROLE_ARN}
```

5. Terapkan aplikasi Anda melalui Helm dengan nama rahasia sebagai masukan dan verifikasi bahwa aplikasi dapat memanggil operasi License Manager API untuk melakukan pemeriksaan hak. Untuk perubahan spesifikasi Helm dan penerapan, lihat Langkah 9 di [Mengintegrasikan produk AWS Marketplace for Containers Anywhere dengan License Manager](#)

Menguji integrasi License Manager di Amazon EKS

Anda juga dapat menguji integrasi License Manager di Amazon EKS. Uji untuk memastikan bahwa aplikasi dapat memanggil operasi License Manager API tanpa rahasia konfigurasi lisensi. Juga pastikan bahwa akun layanan dapat digunakan untuk mengatur Peran IAM untuk Akun Layanan (IRSA) dan memberikan kredensi yang relevan untuk aplikasi.

Untuk menguji integrasi License Manager di Amazon EKS

1. Buat lisensi pengujian di akun penjual uji dengan hak yang diinginkan. Lihat [referensi CreateLicense API](#) untuk menyiapkan lisensi pengujian Anda atau gunakan skrip berikut untuk membuatnya dan membuat hibah lisensi ke akun pembeli pengujian untuk menggunakan lisensi. Skrip berikut menggunakan kredensial akun penjual uji.

```
read -p 'AWS Account for test buyer: ' TEST_BUYER_ACCOUNT_ID
read -p 'License entitlements: ' ENTITLEMENTS

# TEST_SELLER_ACCOUNT_ID="109876543210"
# ENTITLEMENTS="{\"Name\": \"ByData\", \"MaxCount\": 1000, \"Overage\": true, \"Unit\": \"Gigabits\", \"AllowCheckIn\": true}"
```

```

# Create License

NOW=$(date +"%Y-%m-%dT00:00:00+00:00")

PRODUCT_NAME="My awesome product"
PRODUCT_SKU="c97b7825-44c4-4f42-b025-12baa4c171e0"

LICENSE_BENEFICIARY=" arn:aws:iam::$TEST_BUYER_ACCOUNT_ID:root "
LICENSE_ISSUER_NAME="test-seller"
LICENSE_NAME="test-seller-license"

CLIENT_TOKEN="b3920968-a94f-4547-af07-3dd232319367"
CONSUMPTION_TTL=180
CONSUMPTION_RENEW_TYPE="None"

HOME_REGION="us-east-1"

LICENSE_ARN=$(aws license-manager create-license --license-name
"$LICENSE_NAME" --product-name "$PRODUCT_NAME" --product-sku
"$PRODUCT_SKU" --issuer Name="$LICENSE_ISSUER_NAME" --home-region
"$HOME_REGION" --validity Begin="$NOW" --entitlements "$ENTITLEMENTS"
--beneficiary "$LICENSE_BENEFICIARY" --consumption-configuration
RenewType="$CONSUMPTION_RENEW_TYPE",ProvisionalConfiguration={MaxTimeToLiveInMinutes=
$CONSUMPTION_TTL} --client-token "$CLIENT_TOKEN" | jq -r ".LicenseArn" )

echo "License arn: $LICENSE_ARN"

# Create Grant

GRANT_TOKEN="e9a14140-4fca-4219-8230-57511a6ea6"
GRANT_NAME="test-grant"

GRANT_ARN=$(aws license-manager create-grant --grant-name "$GRANT_NAME"
--license-arn "$LICENSE_ARN" --principals "$LICENSE_BENEFICIARY" --home-
region "$HOME_REGION" --client-token "$GRANT_TOKEN" --allowed-operations
"CheckoutLicense" "CheckInLicense" "ExtendConsumptionLicense" "CreateToken" | jq -
r ".GrantArn")

echo "Grant arn: $GRANT_ARN"

```

2. Buat cluster pengujian Amazon EKS dari konfigurasi yang diinginkan, atau jalankan perintah berikut untuk menggunakan konfigurasi default.

```
aws ec2 create-key-pair --region us-west-2 --key-name eks-key-pair
```

```
eksctl create cluster \  
--name awsm-p-eks-test-example \  
--region us-west-2 \  
--with-oidc \  
--ssh-access \  
--ssh-public-key eks-key-pair
```

3. Buat akun layanan untuk klaster yang ada dan kaitkan dengan peran IAM. Perintah berikut menciptakan peran IAM dengan `AWSLicenseManagerConsumptionPolicy`. Kemudian, perintah melampirkannya ke akun `test_sa` layanan cluster Amazon EKS di mana gambar terintegrasi License Manager harus digunakan. Akibatnya, akun layanan bisa mendapatkan kredensi yang sesuai untuk memanggil operasi License Manager API.

```
eksctl create iamserviceaccount \  
--name test_sa \  
--namespace test_namespace \  
--cluster awsm-p-eks-test-example \  
--attach-policy-arn "arn:aws:iam::aws:policy/service-role/  
AWSLicenseManagerConsumptionPolicy" \  
--approve \  
--override-existing-serviceaccounts
```

4. Terapkan aplikasi melalui Helm di akun layanan tempat peran IAM dikaitkan dari perintah sebelumnya. Verifikasi bahwa aplikasi dapat memanggil operasi License Manager API untuk melakukan pemeriksaan hak.

Hak lisensi mengambang dengan License Manager

Dengan lisensi mengambang, saat pengguna masuk ke aplikasi, lisensi diambil dari kumpulan lisensi yang tersedia. Saat pengguna keluar, lisensi ditambahkan kembali ke kumpulan lisensi yang tersedia.

Untuk lisensi mengambang, aplikasi menggunakan operasi `CheckoutLicense` API untuk memeriksa hak dari kumpulan hak saat sumber daya digunakan. Respons operasi `CheckoutLicense` API mencakup token konsumsi lisensi yang merupakan pengidentifikasi unik untuk checkout. Token konsumsi lisensi dapat melakukan tindakan tambahan pada hak yang diperiksa, seperti memverifikasinya kembali ke kumpulan lisensi atau memperpanjang checkout.

Ketika sumber daya tidak lagi digunakan, aplikasi menggunakan operasi `CheckInLicense` API untuk memeriksa hak kembali ke kumpulan.

```
aws license-manager check-in-license \  
--license-consumption-token "f1603b3c1f574b7284db84a9e771ee12"
```

Jika memeriksa lisensi kembali ke pool gagal, misalnya, jika aplikasi mogok selama operasi, hak diperiksa kembali ke kolam secara otomatis setelah 60 menit. Karena itu, jika sumber daya digunakan lebih dari 60 menit, itu adalah praktik terbaik untuk menjaga hak diperiksa dari kolam. Untuk melakukan ini, gunakan operasi `ExtendLicenseConsumption` API selama sumber daya digunakan.

```
aws license-manager extend-license-consumption \  
--license-consumption-token "f1603b3c1f574b7284db84a9e771ee12"
```

Praktik terbaik untuk mengintegrasikan dengan License Manager untuk penerapan lokal

Penerapan aplikasi kontainer di lingkungan lokal mungkin mengalami akses jaringan keluar yang tidak dapat diandalkan. Gunakan praktik terbaik berikut untuk menambah ketahanan guna menghindari gangguan layanan kepada pembeli karena potensi masalah yang disebabkan oleh konektivitas internet yang buruk:

- Coba lagi yang memadai - Masalah jaringan sementara dapat membuat aplikasi Anda tidak terhubung. AWS License Manager Terapkan percobaan ulang hingga 30 menit, dengan mundur eksponensial. Ini dapat membantu menghindari pemadaman jangka pendek atau masalah jaringan.
- Hindari batas keras — Aplikasi yang digunakan dalam cluster yang terhubung dapat secara teratur memeriksa lisensi untuk mengidentifikasi perubahan apa pun karena peningkatan atau pembaruan. Dengan akses keluar yang tidak dapat diandalkan, aplikasi mungkin tidak dapat mengidentifikasi perubahan tersebut. Bila memungkinkan, aplikasi harus menghindari gangguan layanan kepada pembeli karena ketidakmampuan untuk memeriksa lisensi melalui License Manager. Aplikasi dapat kembali pada pengalaman uji coba gratis atau sumber terbuka ketika lisensi kedaluwarsa dan mereka tidak dapat memeriksa apakah lisensi valid.
- Beri tahu pelanggan — Saat menggunakan lisensi yang di-cache, setiap perubahan pada lisensi (termasuk pembaruan atau peningkatan) tidak secara otomatis tercermin pada beban kerja yang sedang berjalan. Beri tahu pelanggan Anda (bahwa mereka harus mengizinkan akses keluar ke aplikasi lagi sementara sehingga aplikasi dapat memperbarui lisensi yang di-cache. Misalnya,

beri tahu pelanggan melalui aplikasi itu sendiri atau melalui dokumentasinya. Demikian pula, ketika kembali ke serangkaian fungsi yang lebih rendah, beri tahu pelanggan bahwa hak mereka habis atau lisensi kedaluwarsa. Kemudian, mereka dapat memilih untuk meningkatkan atau memperbarui.

LicenseManagerCredentialsProvider- Implementasi Java

LicenseCredentialsProvider memperluas rantai penyedia kredensi default AWS SDK untuk penggunaan lokal dengan menambahkan LicenseManagerTokenCredentialsProvider

LicenseCredentialsProvider

```
package com.amazon.awsmp.license;

import software.amazon.awssdk.auth.credentials.AwsCredentials;
import software.amazon.awssdk.auth.credentials.AwsCredentialsProvider;
import software.amazon.awssdk.auth.credentials.AwsCredentialsProviderChain;
import software.amazon.awssdk.auth.credentials.DefaultCredentialsProvider;
import software.amazon.awssdk.auth.credentials.internal.LazyAwsCredentialsProvider;
import software.amazon.awssdk.utils.SdkAutoCloseable;

public class LicenseCredentialsProvider implements AwsCredentialsProvider,
    SdkAutoCloseable {
    private static final LicenseCredentialsProvider CREDENTIALS_PROVIDER = new
    LicenseCredentialsProvider();
    private final LazyAwsCredentialsProvider providerChain;

    private LicenseCredentialsProvider() {
        this.providerChain = createChain();
    }

    public static LicenseCredentialsProvider create() {
        return CREDENTIALS_PROVIDER;
    }

    @Override
    public AwsCredentials resolveCredentials() {
        return this.providerChain.resolveCredentials();
    }

    @Override
    public void close() {
```

```

        this.providerChain.close();
    }

    private LazyAwsCredentialsProvider createChain() {
        return LazyAwsCredentialsProvider.create(() -> {
            AwsCredentialsProvider[] credentialsProviders = new
            AwsCredentialsProvider[]{
                DefaultCredentialsProvider.create(),
                LicenseManagerTokenCredentialsProvider.create()};

            return AwsCredentialsProviderChain.builder().reuseLastProviderEnabled(true)
                .credentialsProviders(credentialsProviders).build();
        });
    }
}

```

LicenseManagerTokenCredentialsProvider

LicenseManagerTokenCredentialsProvider menyediakan kredensial dengan menggunakan License Manager OIDC mengeluarkan token identitas di lingkungan lokal. Anda harus menyertakan kode sumber untuk LicenseCredentialsProvider di classpath aplikasi Anda.

```

package com.amazon.awssmp.license;

import software.amazon.awssdk.auth.credentials.AnonymousCredentialsProvider;
import software.amazon.awssdk.auth.credentials.AwsCredentials;
import software.amazon.awssdk.auth.credentials.AwsCredentialsProvider;
import software.amazon.awssdk.core.SdkSystemSetting;
import software.amazon.awssdk.core.client.config.ClientOverrideConfiguration;
import software.amazon.awssdk.core.retry.RetryPolicyContext;
import software.amazon.awssdk.core.retry.conditions.OrRetryCondition;
import software.amazon.awssdk.core.retry.conditions.RetryCondition;
import software.amazon.awssdk.regions.Region;
import software.amazon.awssdk.regions.providers.DefaultAwsRegionProviderChain;
import software.amazon.awssdk.services.licensemanager.LicenseManagerClient;
import software.amazon.awssdk.services.licensemanager.model.GetAccessTokenRequest;
import software.amazon.awssdk.services.licensemanager.model.GetAccessTokenResponse;
import software.amazon.awssdk.services.sts.StsClient;
import
    software.amazon.awssdk.services.sts.auth.StsAssumeRoleWithWebIdentityCredentialsProvider;
import software.amazon.awssdk.services.sts.model.AssumeRoleWithWebIdentityRequest;
import software.amazon.awssdk.services.sts.model.IdpCommunicationErrorException;
import software.amazon.awssdk.utils.IoUtils;

```

```
import software.amazon.awssdk.utils.SdkAutoCloseable;
import software.amazon.awssdk.utils.StringUtils;
import software.amazon.awssdk.utils.SystemSetting;

import java.io.IOException;
import java.io.InputStream;
import java.io.UncheckedIOException;
import java.nio.file.Files;
import java.nio.file.Path;
import java.nio.file.Paths;
import java.time.Duration;
import java.util.function.Supplier;

public class LicenseManagerTokenCredentialsProvider implements AwsCredentialsProvider,
    SdkAutoCloseable {

    private final StsAssumeRoleWithWebIdentityCredentialsProvider credentialsProvider;
    private final RuntimeException loadException;

    private Path licenseAccessTokenFile;
    private String roleArn;
    private String roleSessionName;
    private StsClient stsClient;
    private LicenseManagerClient lmClient;

    public static LicenseManagerTokenCredentialsProvider create() {
        return new Builder().build();
    }

    @Override
    public AwsCredentials resolveCredentials() {
        if (this.loadException != null) {
            throw this.loadException;
        }
        return this.credentialsProvider.resolveCredentials();
    }

    @Override
    public void close() {
        IoUtils.closeQuietly(this.credentialsProvider, null);
        IoUtils.closeQuietly(this.stsClient, null);
        IoUtils.closeIfCloseable(this.lmClient, null);
    }
}
```

```
private LicenseManagerTokenCredentialsProvider(Builder builder) {
    StsAssumeRoleWithWebIdentityCredentialsProvider credentialsProvider = null;
    RuntimeException loadException = null;

    try {
        this.licenseAccessTokenFile =
Paths.get(StringUtils.trim(LicenseSystemSetting.AWS_WEB_IDENTITY_REFRESH_TOKEN_FILE.getStringValue()));
        this.roleArn = SdkSystemSetting.AWS_ROLE_ARN.getStringValueOrThrow();
        this.roleSessionName =
SdkSystemSetting.AWS_ROLE_SESSION_NAME.getStringValue().orElse("aws-sdk-java-" +
System.currentTimeMillis());
        this.stsClient = builder.stsClient != null ? builder.stsClient :
StsClientFactory.create();
        this.lmClient = builder.lmClient != null ? builder.lmClient :
LicenseManagerClientFactory.create();

        AssumeRoleWithWebIdentityRequest request =
AssumeRoleWithWebIdentityRequest.builder()

.roleArn(this.roleArn).roleSessionName(this.roleSessionName).build();

        Supplier<AssumeRoleWithWebIdentityRequest> supplier = new
AssumeRoleRequestSupplier(request,
            this.licenseAccessTokenFile, this.lmClient);

        credentialsProvider =
StsAssumeRoleWithWebIdentityCredentialsProvider.builder()
            .stsClient(this.stsClient).refreshRequest(supplier).build();
    } catch (RuntimeException ex) {
        loadException = ex;
    }

    this.credentialsProvider = credentialsProvider;
    this.loadException = loadException;
}

public static final class Builder {
    private Path licenseAccessTokenFile;
    private String roleArn;
    private String roleSessionName;
    private StsClient stsClient;
    private LicenseManagerClient lmClient;

    public LicenseManagerTokenCredentialsProvider build() {
```

```
        return new LicenseManagerTokenCredentialsProvider(this);
    }

    public LicenseManagerTokenCredentialsProvider.Builder
licenseAccessTokenFile(Path licenseAccessTokenFile) {
        this.licenseAccessTokenFile = licenseAccessTokenFile;
        return this;
    }

    public LicenseManagerTokenCredentialsProvider.Builder roleArn(String roleArn) {
        this.roleArn = roleArn;
        return this;
    }

    public LicenseManagerTokenCredentialsProvider.Builder roleSessionName(String
roleSessionName) {
        this.roleSessionName = roleSessionName;
        return this;
    }

    public LicenseManagerTokenCredentialsProvider.Builder stsClient(StsClient
stsClient) {
        this.stsClient = stsClient;
        return this;
    }

    public LicenseManagerTokenCredentialsProvider.Builder
lmClient(LicenseManagerClient lmClient) {
        this.lmClient = lmClient;
        return this;
    }
}

private static final class AssumeRoleRequestSupplier implements Supplier {
    private final LicenseManagerClient lmClient;
    private final AssumeRoleWithWebIdentityRequest request;
    private final Path webIdentityRefreshTokenFile;

    AssumeRoleRequestSupplier(final AssumeRoleWithWebIdentityRequest request,
                             final Path
webIdentityRefreshTokenFile,
                             final LicenseManagerClient lmClient) {
        this.lmClient = lmClient;
        this.request = request;
    }
}
```

```
        this.webIdentityRefreshTokenFile = webIdentityRefreshTokenFile;
    }

    public AssumeRoleWithWebIdentityRequest get() {
        return this.request.toBuilder()
            .webIdentityToken(getIdentityToken())
            .build();
    }

    private String getIdentityToken() {
        return refreshIdToken(readRefreshToken(this.webIdentityRefreshTokenFile));
    }

    private String readRefreshToken(Path file) {
        try (InputStream webIdentityRefreshTokenStream =
Files.newInputStream(file)) {
            return IoUtils.toUtf8String(webIdentityRefreshTokenStream);
        } catch (IOException e) {
            throw new UncheckedIOException(e);
        }
    }

    private String refreshIdToken(String licenseRefreshToken) {
        final GetAccessTokenRequest request = GetAccessTokenRequest.builder()
            .token(licenseRefreshToken)
            .build();

        GetAccessTokenResponse response = this.lmClient.getAccessToken(request);
        return response.accessToken();
    }
}

private static final class LicenseManagerClientFactory {
    private static final Duration DEFAULT_API_TIMEOUT = Duration.ofSeconds(30);
    private static final Duration DEFAULT_API_ATTEMPT_TIMEOUT =
Duration.ofSeconds(10);

    public static LicenseManagerClient create() {
        return getLicenseManagerClient();
    }

    private static LicenseManagerClient getLicenseManagerClient() {
        ClientOverrideConfiguration configuration =
ClientOverrideConfiguration.builder()
```

```
        .apiCallTimeout(DEFAULT_API_TIMEOUT)
        .apiCallAttemptTimeout(DEFAULT_API_ATTEMPT_TIMEOUT)
        .build();

    LicenseManagerClient client = LicenseManagerClient.builder()
        .region(configureLicenseManagerRegion())
        .credentialsProvider(AnonymousCredentialsProvider.create())
        .overrideConfiguration(configuration).build();
    return client;
}

private static Region configureLicenseManagerRegion() {
    Region defaultRegion = Region.US_EAST_1;

    Region region;
    try {
        region = (new DefaultAwsRegionProviderChain()).getRegion();
    } catch (RuntimeException ex) {
        region = defaultRegion;
    }
    return region;
}

private static final class StsClientFactory {
    private static final Duration DEFAULT_API_TIMEOUT = Duration.ofSeconds(30);
    private static final Duration DEFAULT_API_ATTEMPT_TIMEOUT =
Duration.ofSeconds(10);

    public static StsClient create() {
        return getStsClient();
    }

    private static StsClient getStsClient() {
        OrRetryCondition retryCondition = OrRetryCondition.create(new
StsRetryCondition(),
            RetryCondition.defaultRetryCondition());

        ClientOverrideConfiguration configuration =
ClientOverrideConfiguration.builder()
            .apiCallTimeout(DEFAULT_API_TIMEOUT)
            .apiCallAttemptTimeout(DEFAULT_API_ATTEMPT_TIMEOUT)
            .retryPolicy(r -> r.retryCondition(retryCondition))
            .build();
```

```
        return StsClient.builder()
            .region(configureStsRegion())
            .credentialsProvider(AnonymousCredentialsProvider.create())
            .overrideConfiguration(configuration).build();
    }

    private static Region configureStsRegion() {
        Region defaultRegion = Region.US_EAST_1;
        Region stsRegion;
        try {
            stsRegion = (new DefaultAwsRegionProviderChain()).getRegion();
        } catch (RuntimeException ex) {
            stsRegion = defaultRegion;
        }
        return stsRegion;
    }

    private static final class StsRetryCondition implements RetryCondition {
        public boolean shouldRetry(RetryPolicyContext context) {
            return context.exception() instanceof IdpCommunicationErrorException;
        }
    }
}

private enum LicenseSystemSetting implements SystemSetting {
    AWS_WEB_IDENTITY_REFRESH_TOKEN_FILE("aws.webIdentityRefreshTokenFile");

    private String systemProperty;
    private String defaultValue = null;

    LicenseSystemSetting(String systemProperty) {
        this.systemProperty = systemProperty;
    }

    @Override
    public String property() {
        return this.systemProperty;
    }

    @Override
    public String environmentVariable() {
        return this.name();
    }
}
```

```

    @Override
    public String defaultValue() {
        return this.defaultValue;
    }
}
}
}

```

LicenseManagerCredentialsProvider- Golang implementasi

LicenseCredentialsProvider

LicenseCredentialsProvider memperluas rantai penyedia kredensi default AWS SDK untuk penggunaan lokal dengan menambahkan `LicenseManagerTokenCredentialsProvider`

```

package lib

import (
    "context"
    "fmt"
    "sync"

    "github.com/aws/aws-sdk-go-v2/aws"
    "github.com/aws/aws-sdk-go-v2/config"
)

// LicenseCredentialsProvider is the custom credential provider that can retrieve valid
// temporary aws credentials
type LicenseCredentialsProvider struct {
    fallbackProvider aws.CredentialsProvider
    mux              sync.RWMutex
    licenseCredentials aws.Credentials
    err              error
}

// NewLicenseCredentialsProvider method will create a LicenseCredentialProvider Object
// which contains valid temporary aws credentials
func NewLicenseCredentialsProvider() (*LicenseCredentialsProvider, error) {
    licenseCredentialProvider := &LicenseCredentialsProvider{}
    fallbackProvider, err := createCredentialProvider()
    if err != nil {
        return licenseCredentialProvider, fmt.Errorf("failed to create
LicenseCredentialsProvider, %w", err)
    }
}

```

```
}
licenseCredentialProvider.fallBackProvider = fallBackProvider
return licenseCredentialProvider, nil
}

// Retrieve method will retrieve temporary aws credentials from the credential provider
func (l *LicenseCredentialsProvider) Retrieve(ctx context.Context) (aws.Credentials,
error) {
l.mux.RLock()
defer l.mux.RUnlock()
l.licenseCredentials, l.err = l.fallBackProvider.Retrieve(ctx)
return l.licenseCredentials, l.err
}

func createCredentialProvider() (aws.CredentialsProvider, error) {
// LoadDefaultConfig will examine all "default" credential providers
ctx := context.TODO()
cfg, err := config.LoadDefaultConfig(ctx)
if err != nil {
return nil, fmt.Errorf("failed to create FallBackProvider, %w", err)
}

var useFallbackProvider bool
if cfg.Credentials != nil {
if _, err := cfg.Credentials.Retrieve(ctx); err != nil {
// If the "default" credentials provider cannot retrieve credentials, enable
fallback to customCredentialsProvider.
useFallbackProvider = true
}
} else {
useFallbackProvider = true
}

if useFallbackProvider {
customProvider, err := newLicenseManagerTokenCredentialsProvider()
if err != nil {
return cfg.Credentials, fmt.Errorf("failed to create fallBackProvider, %w", err)
}
// wrap up customProvider with CredentialsCache to enable caching
cfg.Credentials = aws.NewCredentialsCache(customProvider)
}
return cfg.Credentials, nil
}
```

LicenseManagerTokenCredentialsProvider

LicenseManagerTokenCredentialsProvider menyediakan kredensial dengan menggunakan License Manager OIDC mengeluarkan token identitas di lingkungan lokal. Anda harus menyertakan kode sumber untuk LicenseManagerTokenCredentialsProvider di classpath aplikasi Anda.

```
package lib

import (
    "context"
    "fmt"
    "io/ioutil"
    "os"
    "sync"
    "time"

    "github.com/aws/aws-sdk-go-v2/aws"
    "github.com/aws/aws-sdk-go-v2/config"
    "github.com/aws/aws-sdk-go-v2/service/sts"
)

const awsRefreshTokenFilePathEnvVar = "AWS_LICENSE_ACCESS_FILE"

// licenseManagerTokenCredentialsProvider defines and contains
// StsAssumeRoleWithWebIdentityProvider
type licenseManagerTokenCredentialsProvider struct {
    stsCredentialProvider *stsAssumeRoleWithWebIdentityProvider
    mux                   sync.RWMutex
    licenseCredentials    aws.Credentials
    err                   error
}

// Retrieve method will retrieve credentials from credential provider.
// Make this method public to make this provider satisfies CredentialProvider interface
func (a *licenseManagerTokenCredentialsProvider) Retrieve(ctx context.Context)
(aws.Credentials, error) {
    a.mux.RLock()
    defer a.mux.RUnlock()
    a.licenseCredentials, a.err = a.stsCredentialProvider.Retrieve(ctx)
    return a.licenseCredentials, a.err
}
```

```
// newLicenseManagerTokenCredentialsProvider will create and return
// a LicenseManagerTokenCredentialsProvider Object which wraps up
// stsAssumeRoleWithWebIdentityProvider
func newLicenseManagerTokenCredentialsProvider()
(*licenseManagerTokenCredentialsProvider, error) {
// 1. Retrieve variables From yaml environment
envConfig, err := config.NewEnvConfig()
if err != nil {
return &licenseManagerTokenCredentialsProvider{}, fmt.Errorf("failed to create
LicenseManagerTokenCredentialsProvider, %w", err)
}
roleArn := envConfig.RoleARN
var roleSessionName string
if envConfig.RoleSessionName == "" {
roleSessionName = fmt.Sprintf("aws-sdk-go-v2-%v", time.Now().UnixNano())
} else {
roleSessionName = envConfig.RoleSessionName
}
tokenFilePath := os.Getenv(awsRefreshTokenFilePathEnvVar)
b, err := ioutil.ReadFile(tokenFilePath)
if err != nil {
return &licenseManagerTokenCredentialsProvider{}, fmt.Errorf("failed to create
LicenseManagerTokenCredentialsProvider, %w", err)
}
refreshToken := aws.String(string(b))

// 2. Create stsClient
cfg, err := config.LoadDefaultConfig(context.TODO())
if err != nil {
return &licenseManagerTokenCredentialsProvider{}, fmt.Errorf("failed to create
LicenseManagerTokenCredentialsProvider, %w", err)
}
stsClient := sts.NewFromConfig(cfg, func(o *sts.Options) {
o.Region = configureStsClientRegion(cfg.Region)
o.Credentials = aws.AnonymousCredentials{}
}))

// 3. Configure StsAssumeRoleWithWebIdentityProvider
stsCredentialProvider := newStsAssumeRoleWithWebIdentityProvider(stsClient, roleArn,
roleSessionName, refreshToken)

// 4. Build and return
return &licenseManagerTokenCredentialsProvider{
stsCredentialProvider: stsCredentialProvider,
```

```
    }, nil
  }

func configureStsClientRegion(configRegion string) string {
    defaultRegion := "us-east-1"
    if configRegion == "" {
        return defaultRegion
    } else {
        return configRegion
    }
}
```

Pemberitahuan Amazon SNS untuk produk kontainer

Untuk menerima pemberitahuan, Anda dapat berlangganan topik Amazon Simple Notification Service (Amazon SNS) AWS Marketplace untuk diberikan kepada Anda selama pembuatan produk. Topik memberikan pemberitahuan tentang perubahan langganan pelanggan untuk produk Anda. Misalnya, Anda dapat menggunakan notifikasi ini untuk mengetahui kapan pelanggan menerima penawaran pribadi.

Note

Selama proses pembuatan produk, topik Amazon SNS dibuat untuk produk Anda. Untuk berlangganan notifikasi, Anda memerlukan Nama Sumber Daya Amazon (ARN) dari topik Amazon SNS (misalnya,). `arn:aws:sns:us-east-1:123456789012:aws-mp-subscription-notification-PRODUCTCODE` ARN tidak tersedia di portal penjual untuk produk server. Hubungi [tim AWS Marketplace operasi](#) untuk meminta ARN.

Topik Amazon SNS berikut tersedia untuk produk kontainer:

- [Topik Amazon SNS: `aws-mp-subscription-notification`](#) Topik ini memberi tahu Anda ketika pembeli berlangganan atau berhenti berlangganan suatu produk. Ini tersedia untuk model harga per jam, termasuk per jam dan per jam dengan jangka panjang.

Topik Amazon SNS: `aws-mp-subscription-notification`

Setiap pesan dalam `aws-mp-subscription-notification` topik memiliki format berikut.

```
{
  "action": "<action-name>",
  "customer-identifier": " X01EXAMPLE",
  "product-code": "n0123EXAMPLEXXXXXXXXXXXX",
  "offer-identifier": "offer-abcexample123"
}
```

<action-name> Akan bervariasi tergantung pada notifikasi. Tindakan yang mungkin adalah:

- subscribe-success
- subscribe-fail
- unsubscribe-pending
- unsubscribe-success

offer-identifier ini termasuk dalam pemberitahuan hanya ketika tindakan subscribe-success atau unsubscribe-fail. Itu tidak termasuk dalam notifikasi saat tindakan unsubscribe-pending atau unsubscribe-success. Untuk penawaran yang dibuat sebelum Januari 2024, pengenal ini disertakan dalam pemberitahuan hanya untuk penawaran pribadi. Untuk penawaran yang dibuat pada Januari 2024 dan yang lebih baru, pengenal ini disertakan dalam pemberitahuan untuk semua penawaran, termasuk penawaran pribadi dan penawaran publik.

Untuk informasi tentang jenis penawaran, lihat tanggapan dari [DescribeEntity API](#) atau visibilitas penawaran perjanjian di dasbor [perpanjangan Perjanjian](#).

Note

Untuk [DescribeEntity API](#), jika Anda menemukan aspek penargetan Akun AWS dalam akun dari aturan penargetan untuk penawaran itu, itu adalah penawaran pribadi. Jika tidak ada aspek penargetan Akun AWS dalam akun dari aturan penargetan untuk penawaran itu, itu adalah penawaran umum.

Berlangganan antrian Amazon SQS ke topik Amazon SNS

Sebaiknya berlangganan antrian Amazon SQS ke topik SNS yang disediakan. Untuk petunjuk mendetail tentang cara membuat antrian SQS dan berlangganan antrian ke topik, lihat [Berlangganan antrian Amazon SQS ke topik Amazon SNS di Panduan Pengembang Layanan Pemberitahuan Sederhana Amazon](#).

Note

Anda hanya dapat berlangganan topik AWS Marketplace SNS dari yang Akun AWS digunakan untuk menjual produk. Namun, Anda dapat meneruskan pesan ke akun lain. Untuk informasi selengkapnya, lihat [Mengirim pesan Amazon SNS ke antrean Amazon SQS di akun lain di Panduan Pengembang Layanan Pemberitahuan Sederhana Amazon](#).

Polling antrian SQS untuk pemberitahuan

Setelah Anda berlangganan antrian SQS Anda ke topik SNS, pesan disimpan dalam SQS. Anda harus menentukan layanan yang terus-menerus melakukan polling antrian, mencari pesan, dan menanganinya sesuai dengan itu.

Produk pembelajaran mesin di AWS Marketplace

Sebagai AWS Marketplace penjual, Anda dapat membuat algoritma dan model pembelajaran mesin (ML) yang dapat digunakan pembeli Anda. AWS Topik ini memberikan informasi tentang jenis produk Amazon SageMaker AI yang tercantum di AWS Marketplace.

Ada dua jenis produk SageMaker AI yang tercantum di AWS Marketplace:

Paket model

Model pra-terlatih untuk membuat prediksi yang tidak memerlukan pelatihan lebih lanjut oleh pembeli.

Algoritme

Model yang mengharuskan pembeli untuk menyediakan data pelatihan sebelum membuat prediksi. Algoritma pelatihan disertakan.

Produk-produk ini tersedia untuk pembeli melalui konsol Amazon SageMaker AI atau AWS Marketplace. Pembeli dapat meninjau deskripsi produk, dokumentasi, ulasan pelanggan, harga, dan informasi dukungan. Ketika mereka berlangganan produk paket model atau produk algoritme, itu ditambahkan ke daftar produk mereka di konsol SageMaker AI. Pembeli juga dapat menggunakan AWS SDKs AWS Command Line Interface (AWS CLI), atau konsol SageMaker AI untuk membuat titik akhir inferensi REST yang dikelola sepenuhnya atau melakukan inferensi pada kumpulan data.

Untuk dukungan pembuatan produk pembelajaran mesin dengan Amazon SageMaker AI, hubungi tim [Operasi AWS Marketplace Penjual](#).

Memahami produk pembelajaran mesin

AWS Marketplace mendukung dua jenis produk pembelajaran mesin, menggunakan Amazon SageMaker AI. Kedua jenis, produk paket model dan produk algoritme, menghasilkan model inferensi yang dapat diterapkan untuk membuat prediksi.

SageMaker Paket model AI

Produk [paket model Amazon SageMaker AI](#) berisi model yang telah dilatih sebelumnya. Model pra-terlatih dapat digunakan dalam SageMaker AI untuk membuat kesimpulan atau prediksi secara real

time atau dalam batch. Produk ini berisi komponen inferensi terlatih dengan artefak model, jika ada. Sebagai penjual, Anda dapat melatih model menggunakan SageMaker AI atau membawa model Anda sendiri.

SageMaker Algoritma AI

Pembeli dapat menggunakan produk [algoritme SageMaker AI](#) untuk melakukan beban kerja pembelajaran mesin yang lengkap. Produk algoritma memiliki dua komponen logis: pelatihan dan inferensi. Di SageMaker AI, pembeli menggunakan kumpulan data mereka sendiri untuk membuat pekerjaan pelatihan dengan komponen pelatihan Anda. Ketika algoritma dalam komponen pelatihan Anda selesai, itu menghasilkan artefak model pembelajaran mesin. SageMaker AI menyimpan artefak model di bucket Amazon Simple Storage Service (Amazon S3) pembeli. Di SageMaker AI, pembeli kemudian dapat menerapkan komponen inferensi Anda bersama dengan artefak model yang dihasilkan untuk melakukan inferensi (atau prediksi) secara real time atau dalam batch.

Menerapkan model inferensi

Apakah model inferensi dibuat dari paket model atau algoritma, ada dua metode untuk menerapkannya:

- **Endpoint** — Metode ini menggunakan SageMaker AI untuk menyebarkan model dan membuat titik akhir API. Pembeli dapat menggunakan endpoint ini sebagai bagian dari layanan backend mereka untuk memberi daya pada aplikasi mereka. Saat data dikirim ke titik akhir, SageMaker AI meneruskannya ke wadah model dan mengembalikan hasilnya dalam respons API. Titik akhir dan kontainer terus berjalan hingga dihentikan oleh pembeli.

Note

Pada tahun AWS Marketplace, metode endpoint disebut sebagai inferensi real-time, dan dalam dokumentasi SageMaker AI, itu disebut sebagai layanan hosting. Untuk informasi selengkapnya, lihat [Menerapkan Model di Amazon SageMaker AI](#).

- **Pekerjaan transformasi Batch** — Dalam metode ini, pembeli menyimpan kumpulan data untuk inferensi di Amazon S3. Saat pekerjaan transformasi batch dimulai, SageMaker AI menerapkan model, meneruskan data dari bucket S3 ke wadah model, dan kemudian mengembalikan hasilnya ke bucket Amazon S3. Ketika pekerjaan selesai, SageMaker AI menghentikan pekerjaan. Untuk informasi selengkapnya, lihat [Menggunakan Transformasi Batch](#).

Note

Kedua metode transparan untuk model karena SageMaker AI meneruskan data ke model dan mengembalikan hasil kepada pembeli.

Siklus hidup produk pembelajaran mesin

Produk pembelajaran mesin AWS Marketplace terdiri dari satu atau lebih versi perangkat lunak dan metadata terkait. Konfigurasi produk mencakup properti penting seperti nama, deskripsi, petunjuk penggunaan, harga, kategorisasi, dan kata kunci pencarian.

Proses pembuatan produk pembelajaran mesin

Untuk membuat daftar produk pembelajaran mesin AWS Marketplace, Anda harus menyelesaikan yang berikut ini:

1. [the section called “Mempersiapkan produk Anda di SageMaker AI”](#)
2. [the section called “Cantumkan produk Anda di AWS Marketplace”](#)

Setelah Anda membuat produk pembelajaran mesin, Anda dapat mengedit dan mengelola produk Anda. Untuk informasi selengkapnya, lihat [the section called “Mengelola produk Anda”](#).

Status produk pembelajaran mesin

Produk baru awalnya memiliki visibilitas terbatas, hanya dapat diakses oleh akun yang diizinkan dan pembuat produk. Setelah pengujian dan validasi, Anda dapat mempublikasikan produk Anda untuk membuatnya tersedia di AWS Marketplace katalog untuk semua pembeli. Produk di AWS Marketplace dapat memiliki nilai status berikut:

Status	Definisi
Pementasan	Status ini menunjukkan produk yang tidak lengkap yang masih Anda tambahkan informasinya. Setelah Anda pertama kali menyimpan dan keluar dari pengalaman swalayan, AWS Marketplace buat produk yang tidak dipublika

Status	Definisi
	sikan yang berisi informasi dari langkah-langkah yang telah diselesaikan. Dari status ini, Anda dapat terus menambahkan informasi atau memodifikasi detail yang dikirimkan.
Terbatas	Produk mencapai status ini setelah dikirimkan ke AWS Marketplace dan melewati semua pemeriksaan validasi. Pada titik ini, produk memiliki halaman detail yang hanya dapat diakses oleh akun Anda dan entitas yang diizinkan. Anda dapat melakukan pengujian produk melalui halaman detail ini.
Publik	Saat Anda siap membuat produk Anda terlihat oleh pembeli untuk berlangganan, perbarui visibilitas produk di konsol. Setelah diproses, produk beralih dari status Terbatas ke Publik. Untuk informasi tentang AWS pedoman, lihat the section called “Persyaratan dan praktik terbaik” .
Dibatasi	Untuk mencegah pengguna baru berlangganan produk Anda, Anda dapat membatasinya dengan memperbarui pengaturan visibilitas. Status Terbatas memungkinkan pengguna yang terdaftar diizinkan untuk terus menggunakan produk, tetapi tidak akan lagi terlihat oleh publik atau tersedia untuk pengguna baru.

Untuk informasi lebih lanjut atau dukungan, hubungi [tim Operasi AWS Marketplace Penjual](#).

Harga produk pembelajaran mesin untuk AWS Marketplace

Anda dapat memilih dari beberapa model harga yang tersedia untuk produk Amazon SageMaker AI Anda di AWS Marketplace. Pembeli yang berlangganan produk Anda menjalankannya di SageMaker AI sendiri Akun AWS. Harga untuk pembeli Anda adalah kombinasi dari biaya infrastruktur untuk

sumber daya yang berjalan di mereka Akun AWS dan harga produk yang Anda tetapkan. Bagian berikut memberikan informasi tentang model harga untuk produk SageMaker AI di AWS Marketplace

Topik

- [Harga infrastruktur](#)
- [Harga perangkat lunak](#)

Harga infrastruktur

Pembeli bertanggung jawab atas semua biaya infrastruktur SageMaker AI saat menggunakan produk Anda. Biaya ini ditentukan oleh AWS dan tersedia di halaman [harga Amazon SageMaker AI](#).

Harga perangkat lunak

Anda menentukan harga perangkat lunak yang AWS Marketplace membebankan pembeli untuk menggunakan produk Anda. Anda menetapkan harga dan ketentuan saat menambahkan produk pembelajaran mesin AWS Marketplace.

Semua harga infrastruktur dan perangkat lunak per jenis instans disajikan kepada pembeli di halaman daftar produk AWS Marketplace sebelum pembeli berlangganan.

Topik

- [Harga gratis](#)
- [Harga per jam](#)
- [Harga inferensi](#)
- [Uji coba gratis](#)

Harga gratis

Anda dapat memilih untuk menawarkan produk Anda secara gratis. Dalam hal ini, pembeli hanya membayar biaya infrastruktur.

Harga per jam

Anda dapat menawarkan produk Anda dengan harga per jam per instance perangkat lunak Anda yang berjalan di SageMaker AI. Anda dapat mengenakan harga per jam yang berbeda untuk setiap jenis instans yang dijalankan perangkat lunak Anda. Sementara pembeli menjalankan perangkat

lunak Anda, AWS Marketplace melacak penggunaan dan kemudian menagih pembeli yang sesuai. Penggunaan diprorata hingga menit.

Untuk produk paket model, pembeli dapat menjalankan perangkat lunak Anda dengan dua cara berbeda. Mereka dapat meng-host titik akhir secara terus menerus untuk melakukan inferensi waktu nyata atau menjalankan pekerjaan transformasi batch pada kumpulan data. Anda dapat menetapkan harga yang berbeda untuk kedua cara pembeli dapat menjalankan perangkat lunak Anda.

Untuk produk algoritme, selain menentukan harga untuk melakukan inferensi, seperti yang disebutkan sebelumnya, Anda juga menentukan harga per jam untuk pekerjaan pelatihan. Anda dapat mengenakan harga per jam yang berbeda untuk setiap jenis instans yang didukung gambar pelatihan Anda.

Harga inferensi

Ketika pembeli menjalankan perangkat lunak Anda dengan menghosting titik akhir untuk terus melakukan inferensi waktu nyata, Anda dapat memilih untuk menetapkan harga per inferensi.

Note

Proses transformasi Batch selalu menggunakan harga per jam. Pekerjaan pelatihan untuk produk algoritme juga selalu menggunakan harga per jam. Anda dapat menetapkan harga ini secara independen dari harga inferensi, dan satu sama lain.

Secara default, dengan harga inferensi, AWS Marketplace menagih pembeli Anda untuk setiap pemanggilan titik akhir Anda. Namun, dalam beberapa kasus, perangkat lunak Anda memproses sekumpulan kesimpulan dalam satu pemanggilan (juga dikenal sebagai mini-batch). Untuk penerapan titik akhir, Anda dapat menunjukkan sejumlah inferensi khusus yang AWS Marketplace harus membebaskan biaya kepada pembeli untuk pemanggilan tunggal tersebut. Untuk melakukan ini, sertakan header pengukuran kustom di header respons HTTP dari pemanggilan Anda, seperti pada contoh berikut. Contoh ini menunjukkan doa yang menagih pembeli untuk tiga kesimpulan.

```
X-Amzn-Inference-Metering: {"Dimension": "inference.count", "ConsumedUnits": 3}
```

Note

Untuk harga inferensi, AWS Marketplace hanya menagih pembeli untuk permintaan di mana kode respons HTTP berada 2XX.

Uji coba gratis

Secara opsional, Anda dapat membuat uji coba gratis untuk produk Anda dan menentukan jumlah hari uji coba gratis. Uji coba gratis bisa 5-31 hari. Selama uji coba gratis, pembeli dapat menjalankan perangkat lunak Anda sebanyak yang mereka inginkan dan tidak dikenakan biaya untuk perangkat lunak Anda. Pembeli dikenakan biaya infrastruktur selama uji coba gratis. Setelah uji coba berakhir, mereka dikenakan biaya perangkat lunak normal Anda, bersama dengan biaya infrastruktur.

Ketika pembeli berlangganan produk dengan uji coba gratis, mereka menerima pesan email selamat datang. Pesan tersebut mencakup jangka waktu uji coba gratis, tanggal kedaluwarsa yang dihitung, dan detail tentang berhenti berlangganan. Pesan email pengingat dikirim tiga hari sebelum tanggal kedaluwarsa.

Jika Anda menawarkan uji coba gratis untuk produk Anda di AWS Marketplace, Anda menyetujui [kebijakan pengembalian dana](#) khusus untuk uji coba gratis.

Note

Untuk informasi tentang Penawaran pribadi untuk pembelajaran mesin, lihat [Penawaran pribadi](#).

Pembatasan layanan dan kuota untuk produk pembelajaran mesin di AWS Marketplace

Bagian ini menjelaskan batasan dan kuota pada produk pembelajaran mesin (ML) Anda di AWS Marketplace.

Topik

- [Isolasi jaringan](#)
- [Ukuran gambar](#)
- [Ukuran penyimpanan](#)
- [Ukuran instans](#)
- [Ukuran muatan untuk inferensi](#)
- [Waktu pemrosesan untuk inferensi](#)
- [Kuota layanan](#)

- [Inferensi asinkron](#)
- [Inferensi tanpa server](#)
- [Pelatihan spot terkelola](#)
- [Gambar Docker dan Akun AWS](#)
- [Menerbitkan paket model dari algoritma bawaan atau AWS Marketplace](#)
- [Didukung Wilayah AWS untuk penerbitan](#)

Isolasi jaringan

Untuk tujuan keamanan, saat pembeli berlangganan produk kontainer Anda, kontainer Docker dijalankan di lingkungan yang terisolasi tanpa akses jaringan. Saat Anda membuat kontainer, jangan mengandalkan panggilan keluar melalui internet karena mereka akan gagal. Panggilan ke juga Layanan AWS akan gagal.

Ukuran gambar

[Ukuran gambar Docker Anda diatur oleh kuota layanan Amazon Elastic Container Registry \(Amazon ECR\)](#). Ukuran gambar Docker memengaruhi waktu startup selama pekerjaan pelatihan, pekerjaan transformasi batch, dan pembuatan titik akhir. Untuk kinerja yang lebih baik, pertahankan ukuran gambar Docker yang optimal.

Ukuran penyimpanan

Saat Anda membuat titik akhir, Amazon SageMaker AI akan melampirkan volume penyimpanan Amazon Elastic Block Store (Amazon EBS) ke setiap instans komputasi ML yang menampung titik akhir. (Endpoint juga dikenal sebagai inferensi real-time atau layanan hosting Amazon SageMaker AI.) Ukuran volume penyimpanan tergantung pada jenis instans. Untuk informasi selengkapnya, lihat [Volume Penyimpanan Instans Host](#) di Panduan Pengembang Amazon SageMaker AI.

Untuk transformasi batch, lihat [Penyimpanan dalam Transformasi Batch](#) di Panduan Pengembang Amazon SageMaker AI.

Ukuran instans

SageMaker AI menyediakan pilihan jenis instans yang dioptimalkan agar sesuai dengan kasus penggunaan ML yang berbeda. Jenis instans terdiri dari berbagai kombinasi CPU, GPU, memori, dan kapasitas jaringan. Jenis instans memberi Anda fleksibilitas untuk memilih campuran sumber

daya yang sesuai untuk membangun, melatih, dan menerapkan model ML Anda. Untuk informasi selengkapnya, lihat [Jenis Instans Amazon SageMaker AI ML](#).

Ukuran muatan untuk inferensi

Untuk titik akhir, batasi ukuran maksimum data input per pemanggilan hingga 6 MB. Nilai ini tidak dapat disesuaikan.

Untuk transformasi batch, ukuran maksimum data input per pemanggilan adalah 100 MB. Nilai ini tidak dapat disesuaikan.

Waktu pemrosesan untuk inferensi

Untuk titik akhir, waktu pemrosesan maksimum per pemanggilan adalah 60 detik. Nilai ini tidak dapat disesuaikan.

Untuk transformasi batch, waktu pemrosesan maksimum per pemanggilan adalah 60 menit. Nilai ini tidak dapat disesuaikan.

Kuota layanan

Untuk informasi selengkapnya tentang kuota yang terkait dengan pelatihan dan inferensi, lihat [Amazon SageMaker AI Service Quotas](#).

Inferensi asinkron

Paket model dan algoritme yang diterbitkan di tidak AWS Marketplace dapat diterapkan ke titik akhir yang dikonfigurasi untuk Inferensi Asinkron [Amazon SageMaker AI](#). Titik akhir yang dikonfigurasi untuk inferensi asinkron membutuhkan model untuk memiliki konektivitas jaringan. Semua AWS Marketplace model beroperasi dalam isolasi jaringan. Untuk informasi selengkapnya, lihat [Tidak ada akses jaringan](#).

Inferensi tanpa server

Paket model dan algoritme yang diterbitkan di tidak AWS Marketplace dapat diterapkan ke titik akhir yang dikonfigurasi untuk Inferensi Tanpa Server [Amazon SageMaker AI](#). Titik akhir yang dikonfigurasi untuk inferensi tanpa server memerlukan model untuk memiliki konektivitas jaringan. Semua AWS Marketplace model beroperasi dalam isolasi jaringan. Untuk informasi selengkapnya, lihat [Tidak ada akses jaringan](#).

Pelatihan spot terkelola

Untuk semua algoritma dari AWS Marketplace, nilai `MaxWaitTimeInSeconds` diatur ke 3.600 detik (60 menit), bahkan jika pos pemeriksaan untuk [pelatihan spot terkelola diimplementasikan](#). Nilai ini tidak dapat disesuaikan.

Gambar Docker dan Akun AWS

Untuk penerbitan, gambar harus disimpan di repositori Amazon ECR yang dimiliki oleh penjual Akun AWS. Tidak mungkin mempublikasikan gambar yang disimpan dalam repositori yang dimiliki oleh orang lain. Akun AWS

Menerbitkan paket model dari algoritma bawaan atau AWS Marketplace

Paket model yang dibuat dari pekerjaan pelatihan menggunakan [algoritme bawaan Amazon SageMaker AI](#) atau algoritme dari AWS Marketplace langganan tidak dapat dipublikasikan.

Anda masih dapat menggunakan artefak model dari pekerjaan pelatihan, tetapi gambar inferensi Anda sendiri diperlukan untuk menerbitkan paket model.

Didukung Wilayah AWS untuk penerbitan

AWS Marketplace mendukung penerbitan paket model dan sumber daya algoritme dari Wilayah AWS mana yang berikut ini benar:

- Wilayah yang [didukung Amazon SageMaker AI](#)
- [Wilayah yang tersedia yang diikutsertakan secara default \(misalnya, deskripsikan wilayah kembali\)](#)
"OptInStatus": "opt-in-not-required"

Semua aset yang diperlukan untuk menerbitkan paket model atau produk algoritme harus disimpan di Wilayah yang sama yang Anda pilih untuk dipublikasikan. Ini termasuk yang berikut:

- Paket model dan sumber daya algoritme yang dibuat di Amazon SageMaker AI
- Inferensi dan gambar pelatihan yang diunggah ke repositori Amazon ECR
- Artefak model (jika ada) yang disimpan di Amazon Simple Storage Service dan dimuat secara dinamis selama penerapan model untuk sumber daya paket model
- Uji data untuk inferensi dan validasi pelatihan yang disimpan di Amazon S3

Anda dapat mengembangkan dan melatih produk Anda di Wilayah mana pun yang didukung oleh SageMaker AI. Namun, sebelum Anda dapat mempublikasikan, Anda harus menyalin semua aset ke dan membuat ulang sumber daya di Wilayah yang AWS Marketplace mendukung penerbitan dari.

Keamanan dan kekayaan intelektual dengan Amazon SageMaker AI

Amazon SageMaker AI melindungi kekayaan intelektual dan data pembeli Anda untuk model dan algoritme yang diperoleh AWS Marketplace. Bagian berikut memberikan informasi lebih lanjut tentang cara-cara SageMaker AI melindungi kekayaan intelektual dan keamanan data pelanggan.

Topik

- [Melindungi kekayaan intelektual](#)
- [Tidak ada akses jaringan](#)
- [Keamanan data pelanggan](#)

Melindungi kekayaan intelektual

Saat Anda membuat produk, kode tersebut dikemas dalam gambar kontainer Docker. Untuk informasi lebih lanjut, lihat [Mempersiapkan produk Anda di SageMaker AI](#), nanti di panduan ini. Saat Anda mengunggah gambar kontainer, gambar dan artefak dienkripsi saat transit dan saat istirahat. Gambar juga dipindai untuk kerentanan sebelum dipublikasikan.

Untuk membantu melindungi kekayaan intelektual Anda, SageMaker AI hanya mengizinkan pembeli untuk mengakses produk Anda melalui titik akhir. Layanan AWS Pembeli tidak dapat secara langsung mengakses atau menarik gambar kontainer atau artefak model, mereka juga tidak dapat mengakses infrastruktur yang mendasarinya.

Tidak ada akses jaringan

Tidak seperti model dan algoritme SageMaker AI yang dibuat pembeli, ketika pembeli meluncurkan produk Anda AWS Marketplace, model dan algoritme digunakan tanpa akses jaringan. SageMaker AI menyebarkan gambar di lingkungan tanpa akses ke jaringan atau titik Layanan AWS akhir. Misalnya, image container tidak dapat melakukan panggilan API keluar ke layanan di internet, titik akhir [VPC](#), atau lainnya. Layanan AWS

Keamanan data pelanggan

Produk Anda berjalan di SageMaker AI di dalam pembeli Akun AWS. Jadi, ketika pembeli menggunakan produk Anda untuk melakukan inferensi data, Anda sebagai penjual tidak dapat mengakses data mereka.

Untuk produk algoritme, artefak model dikeluarkan oleh gambar pelatihan Anda setelah setiap pekerjaan pelatihan. Artefak model disimpan di akun pembeli. Artefak model dari pekerjaan pelatihan digunakan saat pembeli menyebarkan model dengan gambar inferensi Anda. Untuk melindungi kekayaan intelektual apa pun yang mungkin terkandung dalam artefak model, enkripsi sebelum mengeluarkannya.

Important

Model keamanan ini mencegah kode Anda mengakses internet selama runtime. Oleh karena itu, kode Anda tidak dapat menggunakan sumber daya atau pustaka dari internet, jadi pakatkan dependensi Anda dalam image container Docker. Ini sangat penting jika Anda memilih untuk mengenkripsi artefak keluaran Anda dari pekerjaan pelatihan. Kunci untuk mengenkripsi dan mendekripsi artefak tidak dapat diakses melalui internet saat runtime. Mereka harus dikemas dengan gambar Anda.

Untuk informasi selengkapnya, lihat [Keamanan di Amazon SageMaker AI](#).

Laporan pembelajaran mesin di AWS Marketplace

AWS Marketplace menghasilkan laporan untuk produk SageMaker AI Amazon Anda yang mencakup data tentang pembeli, keuangan, penggunaan, dan pajak. Semua laporan tersedia Portal Manajemen AWS Marketplace di [halaman Laporan](#). Untuk informasi selengkapnya, lihat [Laporan Penjual](#). Bagian berikut memberikan informasi ringkasan tentang laporan untuk produk pembelajaran mesin.

Topik

- [Laporan bisnis harian](#)
- [Laporan pendapatan bulanan](#)
- [Laporan pencairan](#)
- [Laporan dan analisis lainnya](#)

Laporan bisnis harian

Laporan bisnis harian menyediakan jenis instans, jam penggunaan, pendapatan dari biaya perangkat lunak, dan detail lainnya untuk setiap pembeli dan produk. Pembeli dapat diidentifikasi dengan Akun AWS ID mereka. Untuk informasi selengkapnya, lihat [Laporan bisnis harian](#).

Laporan pendapatan bulanan

Laporan pendapatan bulanan memberi Anda pendapatan bulanan yang telah ditagih kepada pembeli Anda karena menggunakan perangkat lunak Anda. Untuk informasi selengkapnya, lihat [Laporan pendapatan tagihan bulanan](#).

Laporan pencairan

Laporan pencairan bulanan memberikan rincian semua dana yang dikumpulkan atas nama Anda selama periode penyelesaian untuk biaya perangkat lunak Anda. Jumlah total penyelesaian yang tercermin dalam laporan harus sesuai dengan jumlah yang disetorkan ke rekening bank Anda. Untuk informasi lebih lanjut, lihat [Laporan Pencairan](#).

Laporan dan analisis lainnya

Untuk laporan lain yang tersedia, lihat [Laporan penjual](#).

Anda juga dapat membuat laporan kustom menggunakan yang tersedia [Umpan data pengiriman penjual di AWS Marketplace](#) dari AWS Marketplace.

Mempersiapkan produk Anda di SageMaker AI

Sebelum Anda dapat mempublikasikan produk Anda AWS Marketplace, Anda harus menyiapkannya di Amazon SageMaker AI. Ada dua jenis produk SageMaker AI yang tercantum dalam AWS Marketplace: paket model dan algoritma. Untuk informasi selengkapnya, lihat [Produk pembelajaran mesin di AWS Marketplace](#). Topik ini memberikan ikhtisar dari tiga langkah yang diperlukan untuk menyiapkan produk Anda:

1. [Mengemas kode Anda ke dalam gambar untuk produk pembelajaran mesin di AWS Marketplace](#)— Untuk menyiapkan paket model atau produk algoritma, Anda harus membuat gambar kontainer Docker untuk produk Anda.
2. [Mengunggah gambar Anda ke Amazon Elastic Container Registry](#)— Setelah mengemas kode Anda dalam gambar kontainer dan mengujinya secara lokal, unggah gambar dan pindai untuk mengetahui kerentanan yang diketahui. Perbaiki kerentanan apa pun sebelum melanjutkan.

3. [Membuat sumber daya Amazon SageMaker AI Anda](#)— Setelah gambar Anda berhasil dipindai, Anda dapat menggunakannya untuk membuat paket model atau sumber daya algoritme di SageMaker AI.

Mengemas kode Anda ke dalam gambar untuk produk pembelajaran mesin di AWS Marketplace

Produk pembelajaran mesin AWS Marketplace menggunakan Amazon SageMaker AI untuk membuat dan menjalankan logika pembelajaran mesin yang Anda sediakan untuk pembeli. SageMaker AI menjalankan gambar kontainer Docker yang berisi logika Anda. SageMaker AI menjalankan kontainer ini dalam infrastruktur yang aman dan terukur. Untuk informasi selengkapnya, lihat [Keamanan dan kekayaan intelektual dengan Amazon SageMaker AI](#). Bagian berikut memberikan informasi tentang cara mengemas kode Anda ke dalam gambar kontainer Docker untuk SageMaker AI.

Topik

- [Jenis gambar kontainer apa yang saya buat?](#)
- [Membuat gambar paket model](#)
- [Membuat gambar algoritme](#)

Jenis gambar kontainer apa yang saya buat?

Dua jenis gambar kontainer adalah gambar inferensi dan gambar pelatihan.

Untuk membuat produk paket model, Anda hanya perlu gambar inferensi. Untuk petunjuk mendetail, lihat [Membuat gambar paket model](#).

Untuk membuat produk algoritme, Anda memerlukan gambar pelatihan dan inferensi. Untuk petunjuk mendetail, lihat [Membuat gambar algoritme](#).

Untuk mengemas kode dengan benar ke dalam gambar kontainer, wadah harus mematuhi struktur file SageMaker AI. Container harus mengekspos endpoint yang benar untuk memastikan bahwa layanan dapat meneruskan data ke dan dari container Anda. Bagian berikut menjelaskan rincian proses ini.

Important

Untuk tujuan keamanan, ketika pembeli berlangganan produk kontainer Anda, kontainer Docker berjalan di lingkungan yang terisolasi tanpa koneksi internet. Saat Anda membuat kontainer, jangan mengandalkan panggilan keluar melalui internet karena mereka akan gagal. Panggilan ke juga Layanan AWS akan gagal. Untuk informasi selengkapnya, lihat bagian [Keamanan dan kekayaan intelektual dengan Amazon SageMaker AI](#).

Secara opsional, saat membuat gambar inferensi dan pelatihan Anda, gunakan wadah dari [Available Deep Learning Containers Images](#) sebagai titik awal. Gambar sudah dikemas dengan benar dengan kerangka kerja pembelajaran mesin yang berbeda.

Membuat gambar paket model

Paket model Amazon SageMaker AI adalah model pra-terlatih yang membuat prediksi dan tidak memerlukan pelatihan lebih lanjut oleh pembeli. Anda dapat membuat paket model di SageMaker AI dan mempublikasikan produk pembelajaran mesin Anda di AWS Marketplace. Bagian berikut Anda cara membuat paket model untuk AWS Marketplace. Ini termasuk membuat gambar kontainer dan membangun serta menguji gambar secara lokal.

Topik

- [Gambaran Umum](#)
- [Buat gambar inferensi untuk paket model](#)

Gambaran Umum

Paket model mencakup komponen-komponen berikut:

- Gambar inferensi yang disimpan di [Amazon Elastic Container Registry](#) (Amazon ECR)
- (Opsional) Artefak model, disimpan secara terpisah di [Amazon S3](#)

Note

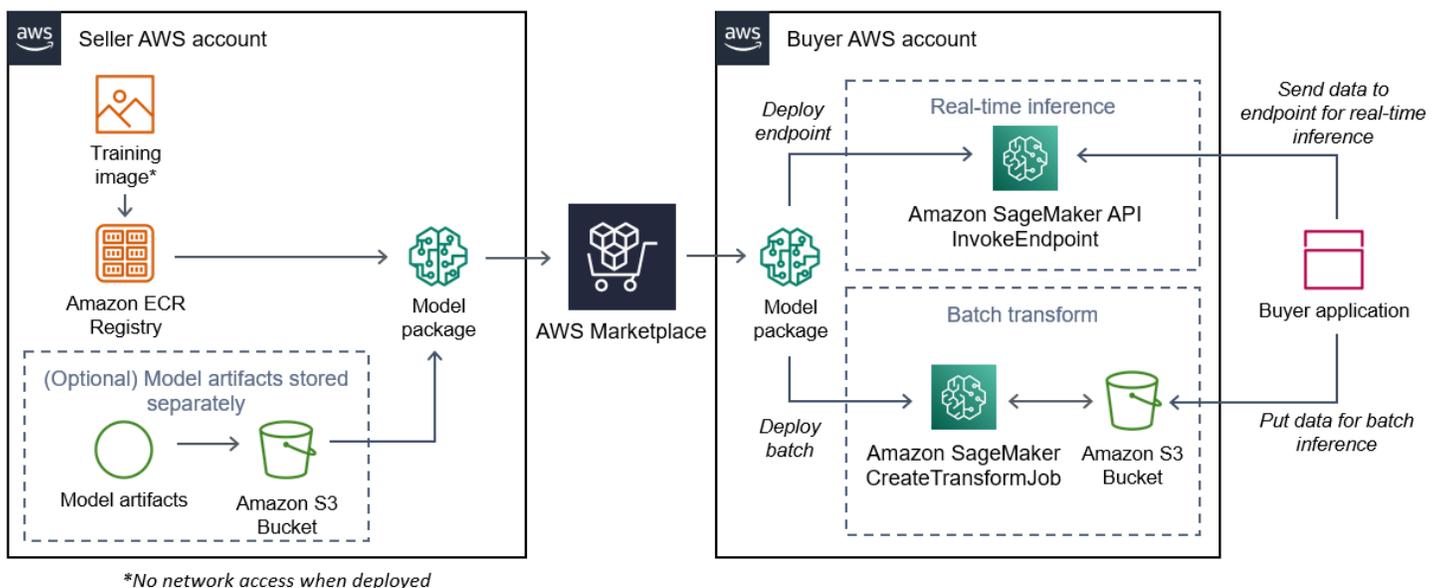
Artefak model adalah file yang digunakan model Anda untuk membuat prediksi dan umumnya merupakan hasil dari proses pelatihan Anda sendiri. Artefak dapat berupa jenis file apa pun yang dibutuhkan oleh model Anda tetapi harus kompresi use.tar.gz. Untuk paket model,

paket tersebut dapat dibundel dalam gambar inferensi Anda atau disimpan secara terpisah di Amazon SageMaker AI. Artefak model yang disimpan di Amazon S3 dimuat ke dalam wadah inferensi saat runtime. Saat mempublikasikan paket model Anda, artefak tersebut diterbitkan dan disimpan dalam bucket Amazon S3 AWS Marketplace milik yang tidak dapat diakses oleh pembeli secara langsung.

Tip

Jika model inferensi Anda dibangun dengan kerangka pembelajaran mendalam seperti Gluon, Keras,,, TensorFlow -Lite MXNet PyTorch TensorFlow, atau ONNX, pertimbangkan untuk menggunakan Amazon AI Neo. SageMaker Neo dapat secara otomatis mengoptimalkan model inferensi yang diterapkan ke keluarga jenis instance cloud tertentu seperti ml.c4,m1.p2, dan lainnya. Untuk informasi selengkapnya, lihat [Mengoptimalkan kinerja model menggunakan Neo](#) di Panduan Pengembang Amazon SageMaker AI.

Diagram berikut menunjukkan alur kerja untuk menerbitkan dan menggunakan produk paket model.



Alur kerja untuk membuat paket model SageMaker AI AWS Marketplace mencakup langkah-langkah berikut:

1. Penjual membuat gambar inferensi (tidak ada akses jaringan saat digunakan) dan mendorongnya ke Amazon ECR Registry.

- Artefak model dapat dibundel dalam gambar inferensi atau disimpan secara terpisah di S3.
2. Penjual kemudian membuat sumber daya paket model di Amazon SageMaker AI dan menerbitkan produk ML-nya. AWS Marketplace
 3. Pembeli berlangganan produk ML dan menyebarkan model.

 Note

Model ini dapat digunakan sebagai titik akhir untuk inferensi waktu nyata atau sebagai pekerjaan batch untuk mendapatkan prediksi untuk seluruh kumpulan data sekaligus. Untuk informasi selengkapnya, lihat [Menerapkan Model untuk Inferensi](#).

4. SageMaker AI menjalankan gambar inferensi. Artefak model apa pun yang disediakan penjual yang tidak dibundel dalam gambar inferensi dimuat secara dinamis saat runtime.
5. SageMaker AI meneruskan data inferensi pembeli ke wadah dengan menggunakan titik akhir HTTP kontainer dan mengembalikan hasil prediksi.

Buat gambar inferensi untuk paket model

Bagian ini memberikan panduan untuk mengemas kode inferensi Anda ke dalam gambar inferensi untuk produk paket model Anda. Prosesnya terdiri dari langkah-langkah berikut:

Langkah-langkah

- [Langkah 1: Buat gambar wadah](#)
- [Langkah 2: Membangun dan menguji gambar secara lokal](#)

Gambar inferensi adalah gambar Docker yang berisi logika inferensi Anda. Container saat runtime mengekspos titik akhir HTTP untuk memungkinkan SageMaker AI meneruskan data ke dan dari container Anda.

 Note

Berikut ini hanya satu contoh kode kemasan untuk gambar inferensi. Untuk informasi selengkapnya, lihat [Menggunakan kontainer Docker dengan SageMaker AWS Marketplace SageMaker AI dan contoh AI](#). GitHub

Contoh berikut menggunakan layanan web, [Flask](#), untuk kesederhanaan, dan tidak dianggap siap produksi.

Langkah 1: Buat gambar wadah

Agar gambar inferensi kompatibel dengan SageMaker AI, image Docker harus mengekspos titik akhir HTTP. Saat container Anda berjalan, SageMaker AI meneruskan input pembeli untuk inferensi ke titik akhir HTTP container. Hasil inferensi dikembalikan di badan respons HTTP.

Panduan berikut menggunakan CLI Docker di lingkungan pengembangan menggunakan distribusi Linux Ubuntu.

- [Buat skrip server web](#)
- [Buat skrip untuk container run](#)
- [Buat Dockerfile](#)
- [Package atau upload artefak model](#)

Buat skrip server web

Contoh ini menggunakan server Python yang disebut [Flask](#), tetapi Anda dapat menggunakan server web apa pun yang berfungsi untuk kerangka kerja Anda.

Note

[Labu](#) digunakan di sini untuk kesederhanaan. Ini tidak dianggap sebagai server web siap produksi.

Buat skrip server web Flask yang melayani dua titik akhir HTTP pada port TCP 8080 yang digunakan AI. SageMaker Berikut ini adalah dua titik akhir yang diharapkan:

- `/ping`— SageMaker AI membuat permintaan HTTP GET ke titik akhir ini untuk memeriksa apakah wadah Anda sudah siap. Ketika penampung Anda siap, ia merespons permintaan HTTP GET di titik akhir ini dengan kode respons HTTP 200.
- `/invocations`— SageMaker AI membuat permintaan HTTP POST ke titik akhir ini untuk inferensi. Data input untuk inferensi dikirim dalam badan permintaan. Jenis konten yang ditentukan

pengguna diteruskan di header HTTP. Tubuh respons adalah output inferensi. Untuk detail tentang batas waktu, lihat [Persyaratan dan praktik terbaik untuk membuat produk pembelajaran mesin](#).

./web_app_serve.py

```
# Import modules
import json
import re
from flask import Flask
from flask import request
app = Flask(__name__)

# Create a path for health checks
@app.route("/ping")
def endpoint_ping():
    return ""

# Create a path for inference
@app.route("/invocations", methods=["POST"])
def endpoint_invocations():

    # Read the input
    input_str = request.get_data().decode("utf8")

    # Add your inference code between these comments.
    #
    #
    #
    #
    # Add your inference code above this comment.

    # Return a response with a prediction
    response = {"prediction": "a", "text": input_str}
    return json.dumps(response)
```

Pada contoh sebelumnya, tidak ada logika inferensi yang sebenarnya. Untuk gambar inferensi Anda yang sebenarnya, tambahkan logika inferensi ke dalam aplikasi web sehingga memproses input dan mengembalikan prediksi yang sebenarnya.

Gambar inferensi Anda harus berisi semua dependensi yang diperlukan karena tidak akan memiliki akses internet, juga tidak akan dapat melakukan panggilan ke salah satu. Layanan AWS

Note

Kode yang sama ini dipanggil untuk inferensi real-time dan batch

Buat skrip untuk container run

Buat skrip bernama `serve` SageMaker AI berjalan saat menjalankan image container Docker. Script berikut memulai server web HTTP.

./serve

```
#!/bin/bash

# Run flask server on port 8080 for SageMaker
flask run --host 0.0.0.0 --port 8080
```

Buat Dockerfile

Buat Dockerfile dalam konteks build Anda. Contoh ini menggunakan Ubuntu 18.04, tetapi Anda dapat memulai dari gambar dasar apa pun yang berfungsi untuk kerangka kerja Anda.

./Dockerfile

```
FROM ubuntu:18.04

# Specify encoding
ENV LC_ALL=C.UTF-8
ENV LANG=C.UTF-8

# Install python-pip
RUN apt-get update \
&& apt-get install -y python3.6 python3-pip \
&& ln -s /usr/bin/python3.6 /usr/bin/python \
&& ln -s /usr/bin/pip3 /usr/bin/pip;

# Install flask server
RUN pip install -U Flask;

# Add a web server script to the image
# Set an environment to tell flask the script to run
```

```
COPY /web_app_serve.py /web_app_serve.py
ENV FLASK_APP=/web_app_serve.py

# Add a script that Amazon SageMaker AI will run
# Set run permissions
# Prepend program directory to $PATH
COPY /serve /opt/program/serve
RUN chmod 755 /opt/program/serve
ENV PATH=/opt/program:${PATH}
```

`Dockerfile` Menambahkan dua skrip yang dibuat sebelumnya ke gambar. Direktori `serve` skrip ditambahkan ke `PATH` sehingga dapat berjalan ketika wadah berjalan.

Package atau upload artefak model

Dua cara untuk menyediakan artefak model mulai dari melatih model hingga gambar inferensi adalah sebagai berikut:

- Dikemas secara statis dengan gambar inferensi.
- Dimuat secara dinamis saat runtime. Karena dimuat secara dinamis, Anda dapat menggunakan gambar yang sama untuk mengemas model pembelajaran mesin yang berbeda.

Jika Anda ingin mengemas artefak model Anda dengan gambar inferensi, sertakan artefak di `Dockerfile`

Jika Anda ingin memuat artefak model secara dinamis, simpan artefak tersebut secara terpisah dalam file terkompresi (`.tar.gz`) di Amazon S3. Saat membuat paket model, tentukan lokasi file terkompresi, dan SageMaker AI mengekstrak dan menyalin konten ke direktori container `/opt/ml/model/` saat menjalankan container Anda. Saat mempublikasikan paket model Anda, artefak tersebut diterbitkan dan disimpan dalam bucket Amazon S3 AWS Marketplace milik yang tidak dapat diakses oleh pembeli secara langsung.

Langkah 2: Membangun dan menguji gambar secara lokal

Dalam konteks build, file berikut sekarang ada:

- `./Dockerfile`
- `./web_app_serve.py`
- `./serve`

- Logika inferensi dan dependensi (opsional) Anda

Selanjutnya membangun, menjalankan, dan menguji image container.

Bangun citra

Jalankan perintah Docker dalam konteks build untuk membangun dan menandai gambar. Contoh ini menggunakan tag `my-inference-image`.

```
sudo docker build --tag my-inference-image ./
```

Setelah menjalankan perintah Docker ini untuk membangun gambar, Anda akan melihat output saat Docker membangun gambar berdasarkan setiap baris di baris Anda. `Dockerfile` Setelah selesai, Anda akan melihat sesuatu yang mirip dengan yang berikut ini.

```
Successfully built abcdef123456  
Successfully tagged my-inference-image:latest
```

Jalankan secara lokal

Setelah build selesai, Anda dapat menguji gambar secara lokal.

```
sudo docker run \  
  --rm \  
  --publish 8080:8080/tcp \  
  --detach \  
  --name my-inference-container \  
  my-inference-image \  
  serve
```

Berikut ini adalah detail tentang perintah:

- `--rm`— Secara otomatis menghapus wadah setelah berhenti.
- `--publish 8080:8080/tcp`— Paparkan port 8080 untuk mensimulasikan port tempat SageMaker AI mengirimkan permintaan HTTP.
- `--detach`— Jalankan wadah di latar belakang.
- `--name my-inference-container`— Beri nama wadah yang sedang berjalan ini.
- `my-inference-image`— Jalankan gambar yang dibangun.

- `serve`— Jalankan skrip yang sama dengan yang dijalankan SageMaker AI saat menjalankan container.

Setelah menjalankan perintah ini, Docker membuat wadah dari gambar inferensi yang Anda buat dan menjalankannya di latar belakang. Wadah menjalankan `serve` skrip, yang meluncurkan server web Anda untuk tujuan pengujian.

Uji titik akhir HTTP ping

Saat SageMaker AI menjalankan wadah Anda, ia secara berkala melakukan ping ke titik akhir. Ketika titik akhir mengembalikan respons HTTP dengan kode status 200, itu memberi sinyal ke SageMaker AI bahwa wadah siap untuk inferensi. Anda dapat menguji ini dengan menjalankan perintah berikut, yang menguji titik akhir dan menyertakan header respons.

```
curl --include http://127.0.0.1:8080/ping
```

Contoh output adalah sebagai berikut.

```
HTTP/1.0 200 OK
Content-Type: text/html; charset=utf-8
Content-Length: 0
Server: MyServer/0.16.0 Python/3.6.8
Date: Mon, 21 Oct 2019 06:58:54 GMT
```

Uji titik akhir HTTP inferensi

Ketika penampung menunjukkan siap dengan mengembalikan kode status 200 ke ping Anda, SageMaker AI meneruskan data inferensi ke titik akhir `/invocations` HTTP melalui permintaan. POST Uji titik inferensi dengan menjalankan perintah berikut.

```
curl \
  --request POST \
  --data "hello world" \
  http://127.0.0.1:8080/invocations
```

Contoh output adalah sebagai berikut.

```
{"prediction": "a", "text": "hello world"}
```

Dengan dua titik akhir HTTP ini berfungsi, gambar inferensi sekarang kompatibel dengan SageMaker AI.

Note

Model produk paket model Anda dapat digunakan dalam dua cara: real time dan batch. Di kedua penerapan, SageMaker AI menggunakan titik akhir HTTP yang sama saat menjalankan wadah Docker.

Untuk menghentikan wadah, jalankan perintah berikut.

```
sudo docker container stop my-inference-container
```

Ketika gambar inferensi Anda siap dan diuji, Anda dapat melanjutkan [Mengunggah gambar Anda ke Amazon Elastic Container Registry](#).

Membuat gambar algoritme

Algoritma Amazon SageMaker AI mengharuskan pembeli membawa data mereka sendiri untuk dilatih sebelum membuat prediksi. Sebagai AWS Marketplace penjual, Anda dapat menggunakan SageMaker AI untuk membuat algoritma pembelajaran mesin (ML) dan model yang dapat digunakan pembeli Anda. AWS Bagian berikut Anda cara membuat gambar algoritma untuk AWS Marketplace. Ini termasuk membuat image pelatihan Docker untuk melatih algoritme Anda dan gambar inferensi yang berisi logika inferensi Anda. Baik gambar pelatihan dan inferensi diperlukan saat menerbitkan produk algoritme.

Topik

- [Gambaran Umum](#)
- [Membuat gambar pelatihan untuk algoritme](#)
- [Membuat gambar inferensi untuk algoritme](#)

Gambaran Umum

Algoritma mencakup komponen-komponen berikut:

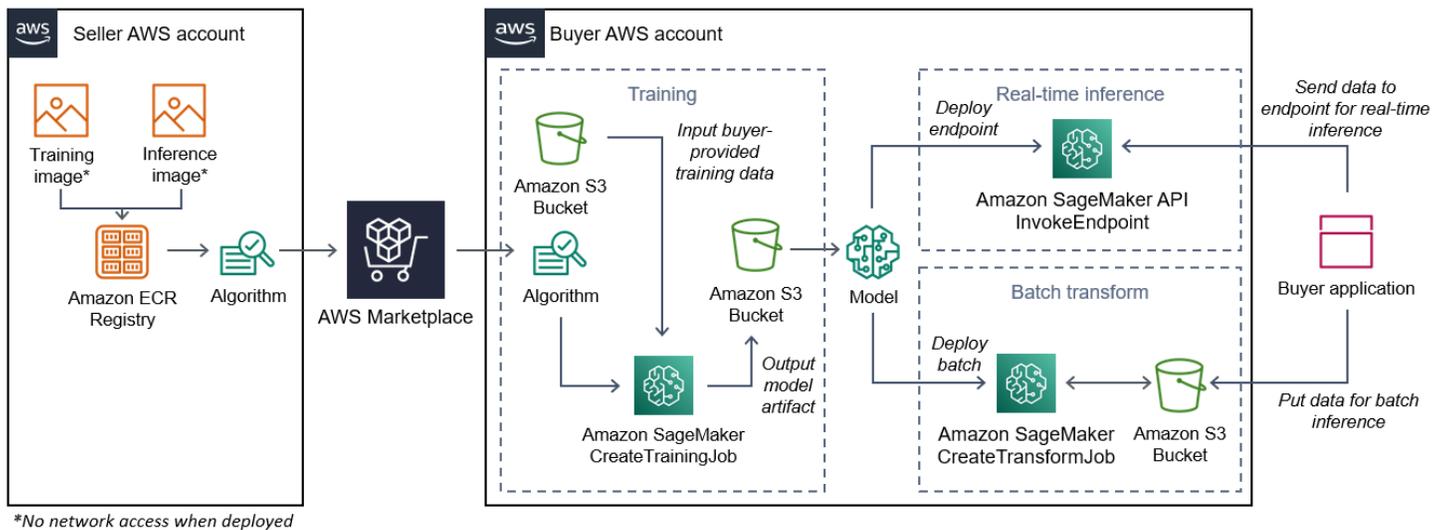
- Gambar pelatihan yang disimpan di [Amazon ECR](#)

- Gambar inferensi yang disimpan di Amazon Elastic Container Registry (Amazon ECR)

Note

Untuk produk algoritme, wadah pelatihan menghasilkan artefak model yang dimuat ke dalam wadah inferensi pada penerapan model.

Diagram berikut menunjukkan alur kerja untuk menerbitkan dan menggunakan produk algoritma.



Alur kerja untuk membuat algoritme SageMaker AI AWS Marketplace mencakup langkah-langkah berikut:

1. Penjual membuat gambar pelatihan dan gambar inferensi (tidak ada akses jaringan saat digunakan) dan mengunggahnya ke Amazon ECR Registry.
2. Penjual kemudian membuat sumber daya algoritme di Amazon SageMaker AI dan menerbitkan produk ML-nya. AWS Marketplace
3. Pembeli berlangganan produk ML.
4. Pembeli membuat pekerjaan pelatihan dengan dataset yang kompatibel dan nilai hyperparameter yang sesuai. SageMaker AI menjalankan gambar pelatihan dan memuat data pelatihan dan hyperparameter ke dalam wadah pelatihan. Ketika pekerjaan pelatihan selesai, artefak model yang terletak di `/opt/ml/model/` dikompresi dan disalin ke ember Amazon S3 pembeli.
5. Pembeli membuat paket model dengan artefak model dari pelatihan yang disimpan di Amazon S3 dan menyebarkan model.

6. SageMaker AI menjalankan gambar inferensi, mengekstrak artefak model terkompresi, dan memuat file ke dalam jalur direktori kontainer inferensi `/opt/ml/model/` di mana ia dikonsumsi oleh kode yang melayani inferensi.
7. Apakah model diterapkan sebagai titik akhir atau pekerjaan transformasi batch, SageMaker AI meneruskan data untuk inferensi atas nama pembeli ke wadah melalui titik akhir HTTP penampung dan mengembalikan hasil prediksi.

 Note

Untuk informasi lebih lanjut, lihat [Model Kereta](#).

Membuat gambar pelatihan untuk algoritme

Bagian ini memberikan panduan untuk mengemas kode pelatihan Anda ke dalam gambar pelatihan. Gambar pelatihan diperlukan untuk membuat produk algoritme.

Gambar pelatihan adalah gambar Docker yang berisi algoritme pelatihan Anda. Wadah mematuhi struktur file tertentu untuk memungkinkan SageMaker AI menyalin data ke dan dari penampung Anda.

Baik gambar pelatihan dan inferensi diperlukan saat menerbitkan produk algoritme. Setelah membuat gambar pelatihan Anda, Anda harus membuat gambar inferensi. Kedua gambar dapat digabungkan menjadi satu gambar atau tetap sebagai gambar terpisah. Apakah akan menggabungkan gambar atau memisahkannya terserah Anda. Biasanya, inferensi lebih sederhana daripada pelatihan, dan Anda mungkin ingin gambar terpisah untuk membantu kinerja inferensi.

 Note

Berikut ini hanya satu contoh kode pengemasan untuk gambar pelatihan. Untuk informasi selengkapnya, lihat [Menggunakan algoritme dan model Anda sendiri dengan contoh AWS Marketplace dan AWS Marketplace SageMaker AI](#). [GitHub](#)

Langkah-langkah

- [Langkah 1: Membuat gambar kontainer](#)
- [Langkah 2: Membangun dan menguji gambar secara lokal](#)

Langkah 1: Membuat gambar kontainer

Agar gambar pelatihan kompatibel dengan Amazon SageMaker AI, gambar harus mematuhi struktur file tertentu agar SageMaker AI dapat menyalin data pelatihan dan input konfigurasi ke jalur tertentu di wadah Anda. Saat pelatihan selesai, artefak model yang dihasilkan disimpan di jalur direktori tertentu di wadah tempat salinan SageMaker AI.

Berikut ini menggunakan CLI Docker yang diinstal di lingkungan pengembangan pada distribusi Ubuntu Linux.

- [Siapkan program Anda untuk membaca input konfigurasi](#)
- [Siapkan program Anda untuk membaca input data](#)
- [Siapkan program Anda untuk menulis output pelatihan](#)
- [Buat skrip untuk container run](#)
- [Buat Dockerfile](#)

Siapkan program Anda untuk membaca input konfigurasi

Jika program pelatihan Anda memerlukan input konfigurasi yang disediakan pembeli, berikut ini adalah di mana program tersebut disalin ke dalam container Anda saat dijalankan. Jika diperlukan, program Anda harus membaca dari jalur file tertentu.

- `/opt/ml/input/config` adalah direktori yang berisi informasi yang mengontrol bagaimana program Anda berjalan.
 - `hyperparameters.json` adalah kamus berformat JSON dari nama dan nilai hyperparameter. Nilainya adalah string, jadi Anda mungkin perlu mengonversinya.
 - `resourceConfig.json` [adalah file berformat JSON yang menjelaskan tata letak jaringan yang digunakan untuk pelatihan terdistribusi](#). Jika gambar pelatihan Anda tidak mendukung pelatihan terdistribusi, Anda dapat mengabaikan file ini.

Note

Untuk informasi selengkapnya tentang input konfigurasi, lihat [Cara Amazon SageMaker AI Menyediakan Informasi Pelatihan](#).

Siapkan program Anda untuk membaca input data

Data pelatihan dapat diteruskan ke wadah dalam salah satu dari dua mode berikut. Program pelatihan Anda yang berjalan di wadah mencerna data pelatihan di salah satu dari dua mode tersebut.

Modus berkas

- `/opt/ml/input/data/<channel_name>` berisi data input untuk saluran itu. Saluran dibuat berdasarkan panggilan ke `CreateTrainingJob` operasi, tetapi umumnya penting bahwa saluran cocok dengan apa yang diharapkan algoritme. File untuk setiap saluran disalin dari [Amazon S3](#) ke direktori ini, mempertahankan struktur pohon yang ditunjukkan oleh struktur kunci Amazon S3.

Modus pipa

- `/opt/ml/input/data/<channel_name>_<epoch_number>` adalah pipa untuk zaman tertentu. Epoch dimulai dari nol dan meningkat satu setiap kali Anda membacanya. Tidak ada batasan jumlah epoch yang dapat Anda jalankan, tetapi Anda harus menutup setiap pipa sebelum membaca epoch berikutnya.

Siapkan program Anda untuk menulis output pelatihan

Output dari pelatihan ditulis ke direktori kontainer berikut:

- `/opt/ml/model` adalah direktori tempat Anda menulis model atau artefak model yang dihasilkan oleh algoritme pelatihan Anda. Model Anda dapat dalam format apa pun yang Anda inginkan. Ini bisa berupa satu file atau seluruh pohon direktori. SageMaker AI mengemas file apa pun di direktori ini ke dalam file terkompresi (`.tar.gz`). File ini tersedia di lokasi Amazon S3 yang dikembalikan oleh operasi `DescribeTrainingJob` API.
- `/opt/ml/output` adalah direktori tempat algoritme dapat menulis `failure` file yang menjelaskan mengapa pekerjaan gagal. Isi file ini dikembalikan di `FailureReason` bidang `DescribeTrainingJob` hasil. Untuk pekerjaan yang berhasil, tidak ada alasan untuk menulis file ini karena diabaikan.

Buat skrip untuk container run

Buat skrip `train shell` yang dijalankan SageMaker AI saat menjalankan image container Docker. Ketika pelatihan selesai dan artefak model ditulis ke direktori masing-masing, keluar dari skrip.

./train

```
#!/bin/bash

# Run your training program here
#
#
#
#
```

Buat Dockerfile

Buat Dockerfile dalam konteks build Anda. Contoh ini menggunakan Ubuntu 18.04 sebagai gambar dasar, tetapi Anda dapat mulai dari gambar dasar apa pun yang berfungsi untuk kerangka kerja Anda.

./Dockerfile

```
FROM ubuntu:18.04

# Add training dependencies and programs
#
#
#
#
# Add a script that SageMaker AI will run
# Set run permissions
# Prepend program directory to $PATH
COPY /train /opt/program/train
RUN chmod 755 /opt/program/train
ENV PATH=/opt/program:${PATH}
```

DockerfileMenambahkan train skrip yang dibuat sebelumnya ke gambar. Direktori skrip ditambahkan ke PATH sehingga dapat berjalan ketika wadah berjalan.

Pada contoh sebelumnya, tidak ada logika pelatihan yang sebenarnya. Untuk gambar pelatihan Anda yang sebenarnya, tambahkan dependensi pelatihan keDockerfile, dan tambahkan logika untuk membaca input pelatihan untuk melatih dan menghasilkan artefak model.

Gambar pelatihan Anda harus berisi semua dependensi yang diperlukan karena tidak akan memiliki akses internet.

Untuk informasi selengkapnya, lihat [Menggunakan algoritme dan model Anda sendiri dengan contoh AWS Marketplace dan AWS Marketplace SageMaker AI](#). [GitHub](#)

Langkah 2: Membangun dan menguji gambar secara lokal

Dalam konteks build, file berikut sekarang ada:

- ./Dockerfile
- ./train
- Ketergantungan dan logika pelatihan Anda

Selanjutnya Anda dapat membangun, menjalankan, dan menguji image container ini.

Bangun citra

Jalankan perintah Docker dalam konteks build untuk membangun dan menandai gambar. Contoh ini menggunakan tagmy-training-image.

```
sudo docker build --tag my-training-image ./
```

Setelah menjalankan perintah Docker ini untuk membangun gambar, Anda akan melihat output saat Docker membangun gambar berdasarkan setiap baris di baris Anda. Dockerfile Setelah selesai, Anda akan melihat sesuatu yang mirip dengan yang berikut ini.

```
Successfully built abcdef123456  
Successfully tagged my-training-image:latest
```

Jalankan secara lokal

Setelah itu selesai, uji gambar secara lokal seperti yang ditunjukkan pada contoh berikut.

```
sudo docker run \  
  --rm \  
  --volume '<path_to_input>:/opt/ml/input:ro' \  
  --volume '<path_to_model>:/opt/ml/model' \  
  --volume '<path_to_output>:/opt/ml/output' \  
  --name my-training-container \  
  my-training-image \  
  train
```

Berikut ini adalah detail perintah:

- `--rm` Secara otomatis menghapus wadah setelah berhenti.
- `--volume '<path_to_input>:/opt/ml/input:ro'`— Jadikan direktori input pengujian tersedia untuk wadah sebagai hanya-baca.
- `--volume '<path_to_model>:/opt/ml/model'`— Bind mount jalur tempat artefak model disimpan di mesin host saat tes pelatihan selesai.
- `--volume '<path_to_output>:/opt/ml/output'`— Bind mount path di mana alasan kegagalan dalam `failure` file ditulis pada mesin host.
- `--name my-training-container`— Beri nama wadah yang sedang berjalan ini.
- `my-training-image`— Jalankan gambar yang dibangun.
- `train`— Jalankan skrip yang sama yang dijalankan SageMaker AI saat menjalankan wadah.

Setelah menjalankan perintah ini, Docker membuat wadah dari gambar pelatihan yang Anda buat dan menjalankannya. Wadah menjalankan `train` skrip, yang memulai program pelatihan Anda.

Setelah program pelatihan Anda selesai dan wadah keluar, periksa apakah artefak model keluaran sudah benar. Selain itu, periksa keluaran log untuk mengonfirmasi bahwa mereka tidak menghasilkan log yang tidak Anda inginkan, sambil memastikan informasi yang cukup diberikan tentang pekerjaan pelatihan.

Ini melengkapi pengemasan kode pelatihan Anda untuk produk algoritme. Karena produk algoritme juga menyertakan gambar inferensi, lanjutkan ke bagian berikutnya, [Membuat gambar inferensi untuk algoritme](#).

Membuat gambar inferensi untuk algoritme

Bagian ini memberikan panduan untuk mengemas kode inferensi Anda ke dalam gambar inferensi untuk produk algoritme Anda.

Gambar inferensi adalah gambar Docker yang berisi logika inferensi Anda. Container saat runtime mengekspos titik akhir HTTP untuk memungkinkan SageMaker AI meneruskan data ke dan dari container Anda.

Baik gambar pelatihan dan inferensi diperlukan saat menerbitkan produk algoritme. Jika Anda belum melakukannya, lihat bagian sebelumnya tentang [Membuat gambar pelatihan untuk algoritme](#). Kedua gambar dapat digabungkan menjadi satu gambar atau tetap sebagai gambar terpisah. Apakah akan menggabungkan gambar atau memisahkannya terserah Anda. Biasanya, inferensi lebih sederhana daripada pelatihan, dan Anda mungkin ingin gambar terpisah untuk membantu kinerja inferensi.

Note

Berikut ini hanya satu contoh kode kemasan untuk gambar inferensi. Untuk informasi selengkapnya, lihat [Menggunakan algoritme dan model Anda sendiri dengan contoh AWS Marketplace dan AWS Marketplace SageMaker AI](#). [GitHub](#)

Contoh berikut menggunakan layanan web, [Flask](#), untuk kesederhanaan, dan tidak dianggap siap produksi.

Langkah-langkah

- [Langkah 1: Membuat gambar inferensi](#)
- [Langkah 2: Membangun dan menguji gambar secara lokal](#)

Langkah 1: Membuat gambar inferensi

Agar gambar inferensi kompatibel dengan SageMaker AI, image Docker harus mengekspos titik akhir HTTP. Saat container Anda berjalan, SageMaker AI meneruskan input untuk inferensi yang diberikan oleh pembeli ke titik akhir HTTP container Anda. Hasil inferensi dikembalikan dalam tubuh respons HTTP.

Berikut ini menggunakan CLI Docker yang diinstal di lingkungan pengembangan pada distribusi Ubuntu Linux.

- [Buat skrip server web](#)
- [Buat skrip untuk container run](#)
- [Buat Dockerfile](#)
- [Mempersiapkan program Anda untuk memuat artefak model secara dinamis](#)

Buat skrip server web

Contoh ini menggunakan server Python yang disebut [Flask](#), tetapi Anda dapat menggunakan server web apa pun yang berfungsi untuk kerangka kerja Anda.

Note

[Labu](#) digunakan di sini untuk kesederhanaan. Ini tidak dianggap sebagai server web siap produksi.

Buat skrip server web Flask yang melayani dua titik akhir HTTP pada port TCP 8080 yang digunakan AI. SageMaker Berikut ini adalah dua titik akhir yang diharapkan:

- `/ping`— SageMaker AI membuat permintaan HTTP GET ke titik akhir ini untuk memeriksa apakah wadah Anda sudah siap. Ketika penampung Anda siap, ia merespons permintaan HTTP GET di titik akhir ini dengan kode respons HTTP 200.
- `/invocations`— SageMaker AI membuat permintaan HTTP POST ke titik akhir ini untuk inferensi. Data input untuk inferensi dikirim dalam badan permintaan. Jenis konten yang ditentukan pengguna diteruskan di header HTTP. Tubuh respons adalah output inferensi.

`./web_app_serve.py`

```
# Import modules
import json
import re
from flask import Flask
from flask import request
app = Flask(__name__)

# Create a path for health checks
@app.route("/ping")
def endpoint_ping():
    return ""

# Create a path for inference
@app.route("/invocations", methods=["POST"])
def endpoint_invocations():

    # Read the input
    input_str = request.get_data().decode("utf8")

    # Add your inference code here.
    #
    #
```

```
#
#
#
# Add your inference code here.

# Return a response with a prediction
response = {"prediction":"a","text":input_str}
return json.dumps(response)
```

Pada contoh sebelumnya, tidak ada logika inferensi yang sebenarnya. Untuk gambar inferensi Anda yang sebenarnya, tambahkan logika inferensi ke dalam aplikasi web sehingga memproses input dan mengembalikan prediksi.

Gambar inferensi Anda harus berisi semua dependensi yang diperlukan karena tidak akan memiliki akses internet.

Buat skrip untuk container run

Buat skrip bernama `serve` SageMaker AI berjalan saat menjalankan image container Docker. Dalam skrip ini, mulai server web HTTP.

./serve

```
#!/bin/bash

# Run flask server on port 8080 for SageMaker AI
flask run --host 0.0.0.0 --port 8080
```

Buat Dockerfile

Buat Dockerfile dalam konteks build Anda. Contoh ini menggunakan Ubuntu 18.04, tetapi Anda dapat memulai dari gambar dasar apa pun yang berfungsi untuk kerangka kerja Anda.

./Dockerfile

```
FROM ubuntu:18.04

# Specify encoding
ENV LC_ALL=C.UTF-8
ENV LANG=C.UTF-8

# Install python-pip
RUN apt-get update \
```

```
&& apt-get install -y python3.6 python3-pip \  
&& ln -s /usr/bin/python3.6 /usr/bin/python \  
&& ln -s /usr/bin/pip3 /usr/bin/pip;  
  
# Install flask server  
RUN pip install -U Flask;  
  
# Add a web server script to the image  
# Set an environment to tell flask the script to run  
COPY /web_app_serve.py /web_app_serve.py  
ENV FLASK_APP=/web_app_serve.py  
  
# Add a script that Amazon SageMaker AI will run  
# Set run permissions  
# Prepend program directory to $PATH  
COPY /serve /opt/program/serve  
RUN chmod 755 /opt/program/serve  
ENV PATH=/opt/program:${PATH}
```

Dockerfile Menambahkan dua skrip yang dibuat sebelumnya ke gambar. Direktori `serve` skrip ditambahkan ke `PATH` sehingga dapat berjalan ketika wadah berjalan.

Mempersiapkan program Anda untuk memuat artefak model secara dinamis

Untuk produk algoritme, pembeli menggunakan kumpulan data mereka sendiri dengan gambar pelatihan Anda untuk menghasilkan artefak model yang unik. Saat proses pelatihan selesai, wadah pelatihan Anda mengeluarkan artefak model ke direktori kontainer. `/opt/ml/model/` SageMaker AI memampatkan konten dalam direktori itu menjadi `file.tar.gz` dan menyimpannya di pembeli di Amazon S3. Akun AWS

Saat model diterapkan, SageMaker AI menjalankan gambar inferensi Anda, mengekstrak artefak model dari `file.tar.gz` yang disimpan di akun pembeli di Amazon S3, dan memuatnya ke wadah inferensi di direktori. `/opt/ml/model/` Saat runtime, kode kontainer inferensi Anda menggunakan data model.

Note

Untuk melindungi kekayaan intelektual apa pun yang mungkin terkandung dalam file artefak model, Anda dapat memilih untuk mengenkripsi file sebelum mengeluarkannya. Untuk informasi selengkapnya, lihat [Keamanan dan kekayaan intelektual dengan Amazon SageMaker AI](#).

Langkah 2: Membangun dan menguji gambar secara lokal

Dalam konteks build, file berikut sekarang ada:

- `./Dockerfile`
- `./web_app_serve.py`
- `./serve`

Selanjutnya Anda dapat membangun, menjalankan, dan menguji image container ini.

Bangun citra

Jalankan perintah Docker untuk membangun dan menandai gambar. Contoh ini menggunakan `my-inference-image`.

```
sudo docker build --tag my-inference-image ./
```

Setelah menjalankan perintah Docker ini untuk membangun gambar, Anda akan melihat output saat Docker membangun gambar berdasarkan setiap baris di baris Anda. Setelah selesai, Anda akan melihat sesuatu yang mirip dengan yang berikut ini.

```
Successfully built abcdef123456  
Successfully tagged my-inference-image:latest
```

Jalankan secara lokal

Setelah build selesai, Anda dapat menguji gambar secara lokal.

```
sudo docker run \  
  --rm \  
  --publish 8080:8080/tcp \  
  --volume '<path_to_model>:/opt/ml/model:ro' \  
  --detach \  
  --name my-inference-container \  
  my-inference-image \  
  serve
```

Berikut ini adalah detail perintah:

- `--rm`- Secara otomatis menghapus wadah setelah berhenti.

- `--publish 8080:8080/tcp`— Ekspos port 8080 untuk mensimulasikan port SageMaker AI mengirimkan permintaan HTTP ke.
- `--volume '<path_to_model>:/opt/ml/model:ro'`— Bind mount path ke tempat artefak model uji disimpan di mesin host sebagai read-only untuk membuatnya tersedia untuk kode inferensi Anda dalam wadah.
- `--detach`— Jalankan wadah di latar belakang.
- `--name my-inference-container`— Beri nama wadah yang sedang berjalan ini.
- `my-inference-image`— Jalankan gambar yang dibangun.
- `serve`— Jalankan skrip yang sama yang dijalankan SageMaker AI saat menjalankan wadah.

Setelah menjalankan perintah ini, Docker membuat wadah dari gambar inferensi dan menjalankannya di latar belakang. Wadah menjalankan `serve` skrip, yang memulai server web Anda untuk tujuan pengujian.

Uji titik akhir HTTP ping

Saat SageMaker AI menjalankan penampung Anda, ia secara berkala melakukan ping ke titik akhir. Ketika titik akhir mengembalikan respons HTTP dengan kode status 200, itu memberi sinyal ke SageMaker AI bahwa wadah siap untuk inferensi.

Jalankan perintah berikut untuk menguji titik akhir dan sertakan header respons.

```
curl --include http://127.0.0.1:8080/ping
```

Contoh output ditunjukkan pada contoh berikut.

```
HTTP/1.0 200 OK
Content-Type: text/html; charset=utf-8
Content-Length: 0
Server: MyServer/0.16.0 Python/3.6.8
Date: Mon, 21 Oct 2019 06:58:54 GMT
```

Uji titik akhir HTTP inferensi

Ketika wadah menunjukkan siap dengan mengembalikan kode status 200, SageMaker AI meneruskan data inferensi ke titik akhir `/invocations` HTTP melalui permintaan. POST

Jalankan perintah berikut untuk menguji titik akhir inferensi.

```
curl \
  --request POST \
  --data "hello world" \
  http://127.0.0.1:8080/invocations
```

Contoh output ditunjukkan pada contoh berikut..

```
{"prediction": "a", "text": "hello world"}
```

Dengan dua titik akhir HTTP ini berfungsi, gambar inferensi sekarang kompatibel dengan SageMaker AI.

Note

Model produk algoritme Anda dapat digunakan dalam dua cara: waktu nyata dan batch. Untuk kedua penerapan, SageMaker AI menggunakan titik akhir HTTP yang sama saat menjalankan container Docker.

Untuk menghentikan kontainer, jalankan perintah berikut.

```
sudo docker container stop my-inference-container
```

Setelah gambar pelatihan dan inferensi Anda untuk produk algoritme Anda siap dan diuji, lanjutkan ke [Mengunggah gambar Anda ke Amazon Elastic Container Registry](#).

Mengunggah gambar Anda ke Amazon Elastic Container Registry

Setelah membuat gambar inferensi dan pelatihan, Anda dapat mengunggahnya ke Amazon Elastic Container Registry. [Amazon ECR adalah registri](#) Docker yang dikelola sepenuhnya. Amazon SageMaker AI menarik gambar dari Amazon ECR untuk membuat paket model untuk inferensi atau algoritma untuk pekerjaan pelatihan. AWS Marketplace juga mengambil gambar-gambar ini dari Amazon ECR untuk mempublikasikan paket model dan produk algoritme Anda. Topik ini memberikan panduan untuk mengunggah inferensi dan gambar pelatihan Anda ke Amazon ECR

Topik

- [Gambar mana yang harus saya unggah?](#)
- [Izin IAM apa yang diperlukan?](#)

- [Log klien Docker Anda ke AWS](#)
- [Buat repositori dan unggah gambar](#)
- [Pindai gambar yang Anda unggah](#)

Gambar mana yang harus saya unggah?

Jika Anda menerbitkan paket model, unggah hanya gambar inferensi. Jika Anda menerbitkan algoritme, unggah gambar inferensi dan gambar pelatihan. Jika gambar inferensi dan pelatihan digabungkan, unggah gambar gabungan hanya sekali.

Izin IAM apa yang diperlukan?

Langkah-langkah berikut mengasumsikan bahwa mesin lokal memiliki AWS kredensial yang benar untuk peran AWS Identity and Access Management (IAM) atau pengguna di penjual. Akun AWS Peran atau pengguna harus memiliki kebijakan yang benar untuk keduanya AWS Marketplace dan Amazon ECR. Misalnya, Anda dapat menggunakan kebijakan AWS terkelola berikut:

- [AWSMarketplaceSellerProductsFullAccess](#)— Untuk akses ke AWS Marketplace
- [AmazonEC2ContainerRegistryFullAccess](#)— Untuk akses ke Amazon ECR

Note

Tautan akan membawa Anda ke Referensi Kebijakan AWS Terkelola.

Log klien Docker Anda ke AWS

Tetapkan variabel untuk Wilayah AWS yang ingin Anda publikasikan dari (lihat [Didukung Wilayah AWS untuk penerbitan](#)). Untuk contoh ini, gunakan Wilayah Timur AS (Ohio).

```
region=us-east-2
```

Jalankan perintah berikut untuk mengatur variabel dengan Akun AWS ID Anda. Contoh ini mengasumsikan bahwa kredensial current AWS Command Line Interface (AWS CLI) milik penjual. Akun AWS

```
account=$(aws sts get-caller-identity --query Account --output text)
```

Untuk mengautentikasi klien CLI Docker Anda dengan registri Akun AWS Amazon ECR Docker untuk Wilayah Anda, jalankan perintah berikut.

```
aws ecr get-login-password \  
--region ${region} \  
| sudo docker login \  
--username AWS \  
--password-stdin \  
${account}.dkr.ecr.${region}.amazonaws.com
```

Buat repositori dan unggah gambar

Tetapkan variabel untuk tag gambar yang diunggah dan variabel lain untuk nama repositori gambar yang diunggah.

```
image=my-inference-image  
repo=my-inference-image
```

Note

Di bagian sebelumnya dari panduan ini di mana gambar inferensi dan pelatihan dibangun, masing-masing diberi tag sebagai *my-inference-image* dan *my-training-image*. Untuk contoh ini, buat dan unggah gambar inferensi ke repositori dengan nama yang sama.

Jalankan perintah berikut untuk membuat repositori gambar di Amazon ECR.

```
aws ecr --region ${region} create-repository --repository-name "${repo}"
```

Nama lengkap lokasi repositori Amazon ECR terdiri dari bagian-bagian berikut: `<account-id>.dkr.ecr.<region>.amazonaws.com/<image-repository-name>`

Untuk mendorong gambar ke repositori, Anda harus menandainya dengan nama lengkap lokasi repositori.

Tetapkan variabel untuk nama lengkap lokasi repositori gambar bersama dengan tag. `latest`

```
fullname="${account}.dkr.ecr.${region}.amazonaws.com/${repo}:latest"
```

Tandai gambar dengan nama lengkap.

```
sudo docker tag ${image} ${fullname}
```

Terakhir, dorong gambar inferensi ke repositori di Amazon ECR.

```
sudo docker push ${fullname}
```

Setelah unggahan selesai, gambar muncul di [daftar repositori konsol Amazon ECR](#) di Wilayah tempat Anda menerbitkan. Pada contoh sebelumnya, gambar didorong ke repositori di Wilayah Timur AS (Ohio).

Pindai gambar yang Anda unggah

Di [konsol Amazon ECR](#), pilih tempat Wilayah AWS Anda menerbitkan, dan buka repositori tempat gambar diunggah. Pilih gambar yang Anda unggah dan mulai pemindaian untuk memeriksa kerentanan yang diketahui. AWS Marketplace memeriksa hasil pemindaian Amazon ECR dari gambar kontainer yang digunakan dalam sumber daya Amazon SageMaker AI Anda sebelum menerbitkannya. Sebelum Anda dapat membuat produk Anda, Anda harus memperbaiki gambar kontainer yang memiliki kerentanan dengan tingkat keparahan Kritis atau Tinggi.

Setelah gambar Anda berhasil dipindai, gambar tersebut dapat digunakan untuk membuat paket model atau sumber daya algoritme.

Jika Anda yakin bahwa produk Anda memiliki kesalahan dalam pemindaian yang positif palsu, hubungi tim [Operasi AWS Marketplace Penjual](#) dengan informasi tentang kesalahan tersebut.

Langkah selanjutnya

- Lihat batas ukuran di [Persyaratan dan praktik terbaik untuk membuat produk pembelajaran mesin](#)
- Lanjutkan ke [Membuat sumber daya Amazon SageMaker AI Anda](#)

Membuat sumber daya Amazon SageMaker AI Anda

Untuk mempublikasikan paket model atau produk algoritme, Anda harus membuat [sumber daya paket model atau sumber daya algoritme](#) masing-masing di Amazon SageMaker AI. Ketika Anda membuat sumber daya Anda untuk suatu AWS Marketplace produk, itu harus disertifikasi melalui langkah validasi. Langkah validasi mengharuskan Anda menyediakan data untuk menguji paket model atau sumber daya algoritme Anda sebelum dapat dipublikasikan. Bagian berikut menunjukkan cara membuat sumber daya SageMaker AI Anda, baik sumber daya paket model atau sumber

daya algoritme. Ini termasuk pengaturan spesifikasi validasi yang memberi tahu SageMaker AI cara melakukan validasi.

Note

Jika Anda belum membuat gambar untuk produk Anda dan mengunggahnya ke Amazon Elastic Container Registry (Amazon ECR) Registry ECR), lihat [Mengemas kode Anda ke dalam gambar untuk produk pembelajaran mesin di AWS Marketplace](#) dan [Mengunggah gambar Anda ke Amazon Elastic Container Registry](#) untuk informasi tentang cara melakukannya.

Topik

- [Membuat paket model Anda](#)
- [Membuat algoritme Anda](#)

Membuat paket model Anda

Berikut ini adalah persyaratan untuk membuat paket model untuk AWS Marketplace:

- Gambar inferensi yang disimpan di [Amazon](#) ECR
- (Opsional) Artefak model, disimpan secara terpisah di [Amazon](#) S3
- Data pengujian Anda yang digunakan untuk inferensi, disimpan di Amazon Simple Storage Service

Note

Berikut ini adalah tentang membuat produk paket model. Untuk informasi selengkapnya tentang paket model di SageMaker AI, lihat [Membuat Sumber Daya Paket Model](#).

Membuat sumber daya paket model

Prosedur berikut mengarahkan Anda melalui pembuatan sumber daya paket model.

Langkah 1: Untuk membuat sumber daya paket model

1. Buka [konsol Amazon SageMaker AI](#).

2. Pastikan Anda berada di AWS Wilayah yang ingin Anda publikasikan dengan melihat di kanan atas halaman. Untuk penerbitan, lihat [Didukung Wilayah AWS untuk penerbitan](#) bagian. Gambar inferensi yang Anda unggah ke Amazon ECR pada langkah sebelumnya harus berada di Wilayah yang sama.
3. Di menu navigasi kiri, pilih Paket model.
4. Pilih Buat paket model.

Setelah Anda membuat paket, Anda perlu mengatur spesifikasi paket inferensi.

Langkah 2: Untuk mengatur spesifikasi inferensi

1. Berikan Nama untuk paket model Anda (misalnya, *my-model-package*).
2. Untuk Lokasi gambar inferensi, masukkan URI gambar inferensi Anda yang diunggah ke Amazon ECR. Anda dapat mengambil URI dengan menemukan gambar Anda di konsol [Amazon ECR](#).
3. Jika artefak model Anda dari pelatihan dibundel dengan logika Anda dalam gambar inferensi Anda, biarkan Lokasi artefak data model kosong. Jika tidak, tentukan lokasi Amazon S3 lengkap dari file terkompresi (.tar.gz) artefak model Anda.
4. Dengan menggunakan kotak tarik-turun, pilih jenis instans yang didukung dari gambar inferensi Anda untuk inferensi real-time (juga dikenal sebagai titik akhir) dan pekerjaan transformasi batch.
5. Pilih Berikutnya.

Sebelum paket model Anda dapat dibuat dan dipublikasikan, validasi diperlukan untuk memastikan bahwa itu berfungsi seperti yang diharapkan. Ini mengharuskan Anda menjalankan pekerjaan transformasi batch dengan data pengujian untuk inferensi yang Anda berikan. Spesifikasi validasi memberi tahu SageMaker AI cara melakukan validasi.

Langkah 3: Untuk mengatur spesifikasi validasi

1. Setel Publikasikan paket model ini AWS Marketplace ke Ya. Jika Anda menyetel ini ke Tidak, Anda tidak dapat mempublikasikan paket model ini nanti. Memilih Ya [mensertifikasi](#) paket model Anda AWS Marketplace dan memerlukan langkah validasi.
2. Jika ini adalah pertama kalinya menyelesaikan proses ini, pilih Buat peran baru untuk peran IAM. Amazon SageMaker AI menggunakan peran ini saat menyebarkan paket model Anda. Ini termasuk tindakan, seperti menarik gambar dari Amazon ECR dan artefak dari Amazon S3.

Tinjau pengaturan, dan pilih Buat peran. Membuat peran di sini memberikan izin yang dijelaskan oleh kebijakan [AmazonSageMakerFullAccessIAM](#) ke peran yang Anda buat.

3. Edit JSON di profil validasi. Untuk detail tentang nilai yang diizinkan, lihat [TransformJobDefinition](#).
 1. `TransformInput.DataSource.S3Uri`: Setel ke tempat data pengujian Anda untuk inferensi disimpan.
 2. `TransformInput.ContentType`: Tentukan jenis konten data pengujian Anda (misalnya `application/json`, `text/plain`, `image/png` , atau nilai lainnya). SageMaker AI tidak memvalidasi data input yang sebenarnya. Nilai ini diteruskan ke titik akhir HTTP kontainer Anda di nilai `Content-type` header.
 3. `TransformInput.CompressionType`: Setel ke `None` jika data pengujian untuk inferensi di Amazon S3 tidak dikompresi.
 4. `TransformInput.SplitType`: Setel `None` untuk meneruskan setiap objek di Amazon S3 secara keseluruhan untuk inferensi.
 5. `TransformOutput.S3OutputPath`: Setel ke lokasi dimana output inferensi disimpan.
 6. `TransformOutput.AssembleWith`: Setel `None` untuk menampilkan setiap inferensi sebagai objek terpisah di Amazon S3.
4. Pilih Buat paket model.

SageMaker AI menarik gambar inferensi dari Amazon ECR, menyalin artefak apa pun ke wadah inferensi, dan menjalankan tugas transformasi batch menggunakan data pengujian Anda untuk inferensi. Setelah validasi berhasil, status berubah menjadi Selesai.

 Note

Langkah validasi tidak mengevaluasi keakuratan model dengan data pengujian Anda. Langkah validasi memeriksa apakah kontainer berjalan dan merespons seperti yang diharapkan.

Anda telah menyelesaikan pembuatan sumber daya produk model Anda. Lanjutkan ke [Cantumkan produk Anda di AWS Marketplace](#).

Membuat algoritme Anda

Berikut ini adalah persyaratan untuk membuat algoritma untuk AWS Marketplace:

- Gambar inferensi, disimpan di Amazon ECR
- Gambar pelatihan, disimpan di Amazon ECR
- Data pengujian Anda untuk pelatihan, disimpan di Amazon S3
- Data pengujian Anda untuk inferensi, disimpan di Amazon S3

Note

Panduan berikut menciptakan produk algoritme. Untuk informasi selengkapnya, lihat [Membuat Sumber Algoritma](#).

Membuat sumber daya algoritme

Prosedur berikut mengarahkan Anda melalui pembuatan sumber daya dalam paket algoritme Anda.

Langkah 1: Untuk membuat sumber daya algoritme

1. Buka [konsol Amazon SageMaker AI](#).
2. Pastikan Anda berada di AWS Wilayah yang ingin Anda publikasikan dengan melihat di kanan atas halaman (lihat [Didukung Wilayah AWS untuk penerbitan](#)). Gambar pelatihan dan inferensi yang Anda unggah ke Amazon ECR pada langkah sebelumnya harus berada di Wilayah yang sama ini.
3. Di menu navigasi kiri, pilih Algoritma.
4. Pilih Buat algoritma.

Setelah Anda membuat paket algoritme, Anda harus mengatur spesifikasi untuk pelatihan dan penyetelan model Anda.

Langkah 2: Untuk mengatur spesifikasi pelatihan dan penyetelan

1. Masukkan Nama untuk algoritma Anda (misalnya, *my-algorithm*).

2. Untuk gambar Pelatihan, tempel lokasi URI lengkap gambar pelatihan Anda yang diunggah ke Amazon ECR. Anda dapat mengambil URI dengan menemukan gambar Anda di konsol [Amazon ECR](#).
3. Dengan menggunakan kotak tarik-turun, pilih jenis instans untuk pelatihan yang didukung gambar pelatihan Anda.
4. Di bawah bagian Spesifikasi saluran, tambahkan saluran untuk setiap kumpulan data input yang didukung algoritme Anda, hingga 20 saluran sumber input. Untuk informasi selengkapnya, lihat [Konfigurasi Data Input](#).
5. Pilih Berikutnya.
6. Jika algoritme Anda mendukung hyperparameters dan tuning hyperparameter, Anda harus menentukan parameter tuning.
7. Pilih Berikutnya.

 Note

Kami sangat menyarankan agar algoritme Anda mendukung penyetelan hyperparameter dan membuat parameter yang sesuai dapat disetel. Hal ini memungkinkan ilmuwan data untuk menyetel model untuk mendapatkan hasil terbaik.

Setelah Anda mengatur parameter penyetelan, jika ada, Anda harus mengatur spesifikasi untuk gambar inferensi Anda.

Langkah 3: Untuk mengatur spesifikasi gambar inferensi

1. Untuk Lokasi gambar inferensi, tempel URI gambar inferensi yang diunggah ke Amazon ECR. Anda dapat mengambil URI dengan menemukan gambar Anda di [Amazon ECR](#) Console.
2. Menggunakan kotak tarik-turun, pilih jenis instans yang didukung untuk gambar inferensi Anda untuk inferensi real-time (juga dikenal sebagai titik akhir) dan pekerjaan transformasi batch.
3. Pilih Berikutnya.

Sebelum algoritma Anda dapat dibuat dan dipublikasikan, validasi diperlukan untuk memastikan bahwa itu berfungsi seperti yang diharapkan. Ini mengharuskan Anda menjalankan pekerjaan pelatihan dengan data pengujian untuk pelatihan dan pekerjaan transformasi batch dengan data

pengujian untuk inferensi yang Anda berikan. Spesifikasi validasi memberi tahu SageMaker AI cara melakukan validasi.

Langkah 4: Untuk mengatur spesifikasi validasi

1. Setel Publikasikan algoritma ini AWS Marketplace ke Ya. Jika Anda menyetel ini ke Tidak, Anda tidak dapat mempublikasikan algoritme ini nanti. Memilih Ya [mensertifikasi](#) algoritme Anda AWS Marketplace dan memerlukan spesifikasi validasi.
2. Jika ini adalah pertama kalinya Anda membuat paket pembelajaran mesin AWS Marketplace, pilih Buat peran baru untuk peran IAM. Amazon SageMaker AI menggunakan peran ini saat melatih algoritme Anda dan menerapkan paket model berikutnya. Ini termasuk tindakan seperti menarik gambar dari Amazon ECR, menyimpan artefak di Amazon S3, dan menyalin data pelatihan dari Amazon S3. Tinjau pengaturan, dan pilih Buat peran. Membuat peran di sini memberikan izin yang dijelaskan oleh kebijakan [AmazonSageMakerFullAccessIAM](#) ke peran yang Anda buat.
3. Edit file JSON di profil validasi untuk definisi pekerjaan Pelatihan. Untuk informasi selengkapnya tentang nilai yang diizinkan, lihat [TrainingJobDefinition](#).
 1. `InputDataConfig`: Dalam array JSON ini, tambahkan [objek Channel](#) untuk setiap saluran yang Anda tentukan dalam langkah spesifikasi pelatihan. Untuk setiap saluran, tentukan di mana data pengujian Anda untuk pelatihan disimpan.
 2. `OutputDataConfig`: Setelah pelatihan selesai, artefak model di jalur `/opt/ml/model/` direktori wadah pelatihan dikompresi dan disalin ke Amazon S3. Tentukan lokasi Amazon S3 tempat file terkompresi (`.tar.gz`) disimpan.
4. Edit file JSON di profil validasi untuk Transform definisi pekerjaan. Untuk informasi selengkapnya tentang nilai yang diizinkan, lihat [TransformJobDefinition](#).
 1. `TransformInput.DataSource.S3Uri`: Setel ke tempat data pengujian Anda untuk inferensi disimpan.
 2. `TransformInput.ContentType`: Tentukan jenis konten data pengujian Anda. Misalnya, `application/json`, `text/plain`, `image/png`, atau nilai lainnya. Amazon SageMaker AI tidak memvalidasi data input aktual. Nilai ini diteruskan ke titik akhir HTTP kontainer Anda di nilai `Content-type` header.
 3. `TransformInput.CompressionType`: Setel ke `None` jika data pengujian untuk inferensi di Amazon S3 tidak dikompresi.

4. `TransformInput.SplitType`: Pilih bagaimana Anda ingin objek di S3 split. Misalnya, `None` melewati setiap objek di Amazon S3 secara keseluruhan untuk inferensi. Untuk detail selengkapnya, lihat [SplitType](#) di Referensi API Amazon SageMaker AI.
 5. `TransformOutput.S3OutputPath`: Setel ke lokasi di mana output inferensi disimpan.
 6. `TransformOutput.AssembleWith`: Setel `None` untuk menampilkan setiap inferensi sebagai objek terpisah di Amazon S3.
5. Pilih Buat paket algoritma.

SageMaker AI menarik gambar pelatihan dari Amazon ECR, menjalankan pekerjaan uji pelatihan menggunakan data Anda, dan menyimpan artefak model di Amazon S3. Kemudian menarik gambar inferensi dari Amazon ECR, menyalin artefak dari Amazon S3 ke dalam wadah inferensi, dan menjalankan tugas transformasi batch menggunakan data pengujian Anda untuk inferensi. Setelah validasi berhasil, status berubah menjadi Selesai.

Note

Langkah validasi tidak mengevaluasi keakuratan pelatihan atau model dengan data pengujian Anda. Langkah validasi memeriksa apakah kontainer berjalan dan merespons seperti yang diharapkan.

Langkah validasi hanya memvalidasi pemrosesan batch. Terserah Anda untuk memvalidasi bahwa pemrosesan waktu nyata bekerja dengan produk Anda.

Anda telah menyelesaikan pembuatan sumber daya produk algoritme Anda. Lanjutkan ke [Cantumkan produk Anda di AWS Marketplace](#).

Cantumkan produk Anda di AWS Marketplace

Setelah Anda mengemas kode Anda ke dalam gambar paket model atau gambar algoritme, mengunggah gambar Anda, dan membuat sumber daya SageMaker AI Amazon, Anda dapat mempublikasikan produk pembelajaran mesin Anda di AWS Marketplace. Bagian berikut memandu Anda melalui proses penerbitan, yang meliputi membuat daftar produk Anda, menguji produk Anda, dan menandatangani untuk penerbitan. Setelah produk Anda diterbitkan, Anda meminta perubahan untuk memperbarui iklan Anda. Lihat informasi yang lebih lengkap di [the section called “Mengelola produk Anda”](#).

Topik

- [Prasyarat](#)
- [Langkah 1: Buat daftar baru](#)
- [Langkah 2: Berikan informasi produk](#)
- [Langkah 3: Tambahkan versi produk awal](#)
- [Langkah 4: Konfigurasi model harga](#)
- [Langkah 5: Konfigurasi kebijakan pengembalian dana](#)
- [Langkah 6: Konfigurasi EULA](#)
- [Langkah 7: Konfigurasi allowlist](#)

Prasyarat

Sebelum Anda dapat mempublikasikan paket model atau algoritme Anda AWS Marketplace, Anda harus memiliki yang berikut:

- An Akun AWS yang terdaftar sebagai AWS Marketplace penjual. Anda dapat melakukan ini di [Portal Manajemen AWS Marketplace](#).
- Profil penjual lengkap di bawah halaman [Pengaturan](#) di Portal Manajemen AWS Marketplace.
- Untuk menerbitkan produk berbayar, Anda harus melengkapi wawancara pajak dan formulir bank. Ini tidak diperlukan untuk menerbitkan produk gratis. Untuk informasi selengkapnya, lihat [Proses pendaftaran Penjual](#).
- Anda harus memiliki izin untuk mengakses Portal Manajemen AWS Marketplace dan Amazon SageMaker AI. Untuk informasi selengkapnya, lihat [Izin yang diperlukan](#).

Izin yang diperlukan

Untuk mempublikasikan produk Amazon SageMaker AI, Anda harus menentukan ARN peran IAM yang valid yang memiliki hubungan kepercayaan dengan kepala layanan AWS Marketplace . Selain itu, pengguna IAM atau peran yang Anda masuki memerlukan izin yang diperlukan.

Mengatur izin masuk

- Tambahkan izin berikut ke peran IAM:
 1. sagemaker: DescribeModelPackage — Untuk daftar paket model

2. sagemaker: DescribeAlgorithm — Untuk daftar algoritma

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "sagemaker:DescribeModelPackage",
        "sagemaker:DescribeAlgorithm"
      ],
      "Resource": "*"
    }
  ]
}
```

Mengatur produk peran AddVersion/Create IAM

1. Ikuti langkah-langkah untuk membuat peran dengan kebijakan kepercayaan khusus. Untuk informasi selengkapnya, lihat [Membuat peran IAM menggunakan kebijakan kepercayaan khusus \(konsol\)](#).
2. Masukkan pernyataan kebijakan kepercayaan kustom berikut ini:

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Statement1",
      "Effect": "Allow",
      "Principal": {
        "Service": "assets.marketplace.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

3. Masukkan kebijakan izin berikut:

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "sagemaker:DescribeModelPackage",
        "sagemaker:DescribeAlgorithm"
      ],
      "Resource": "*"
    }
  ]
}
```

4. Berikan peran ARN saat diminta. Peran harus mengikuti format: `arn:aws:iam::<account-id>:role/<role-name>`.

Untuk AWS Marketplace izin yang diperlukan, atau untuk mengelola akun penjual, lihat [Kebijakan dan izin untuk AWS Marketplace](#) penjual.

Aset yang dibutuhkan

Sebelum membuat daftar produk pembelajaran mesin, pastikan Anda memiliki aset wajib berikut:

- Nama Sumber Daya Amazon (ARN) — Menyediakan ARN dari paket model atau sumber daya algoritme di tempat Wilayah AWS Anda menerbitkan (lihat). [Didukung Wilayah AWS untuk penerbitan](#)
- ARN untuk paket model memiliki formulir ini: `arn:aws:sagemaker:<region>:<account-id>:model-package/<model-package-name>`

Untuk menemukan paket model ARN Anda, lihat Paket [model pasar saya](#).

- ARN untuk algoritma memiliki bentuk ini: `arn:aws:sagemaker:<region>:<account-id>:algorithm/<algorithm-name>`

Untuk menemukan ARN sumber daya algoritme Anda, lihat Algoritma [saya](#).

- [the section called “Persyaratan untuk informasi penggunaan”](#)— Berikan detail tentang input, output, dan contoh kode.
- [the section called “Persyaratan untuk input dan output”](#)— Berikan file atau teks.
- [the section called “Persyaratan untuk notebook Jupyter”](#)— Menunjukkan penggunaan produk lengkap.

Langkah 1: Buat daftar baru

Untuk memulai produk pembelajaran mesin, Anda akan memulai proses listing dengan menyetel nama produk, menambahkan tag sumber daya opsional untuk organisasi, dan membuat ID produk. ID produk digunakan untuk melacak produk Anda sepanjang siklus hidupnya.

Note

Sebelum membuat daftar Anda, pastikan Anda memiliki sumber daya yang diperlukan yang ditentukan [Persyaratan dan praktik terbaik untuk membuat produk pembelajaran mesin](#).

1. Masuk ke penjual Anda Akun AWS dan pergi ke [Portal Manajemen AWS Marketplace](#).
2. Di menu atas, buka Produk dan kemudian pilih Pembelajaran mesin.
3. Pilih Buat produk pembelajaran mesin.
4. Di bawah Nama Produk, masukkan nama produk unik yang akan ditampilkan kepada pembeli di bagian atas halaman daftar produk dan di hasil pencarian.
5. (Opsional) Di bawah Tag, masukkan tag apa pun yang ingin Anda kaitkan dengan produk. Untuk informasi selengkapnya, lihat [Menandai AWS sumber daya](#).
6. Di bawah Product ID dan kode, pilih Hasilkan ID produk dan kode.
7. Pilih Lanjutkan ke wizard. Anda akan memulai proses menambahkan informasi produk terperinci di wizard.

Langkah 2: Berikan informasi produk

Saat mencantumkan produk pembelajaran mesin Anda AWS Marketplace, memberikan informasi produk yang komprehensif dan akurat sangat penting. Gunakan langkah Menyediakan informasi produk di wizard untuk menangkap detail penting tentang penawaran Anda seperti kategori produk dan informasi dukungan.

1. Masukkan informasi tentang produk Anda.
2. Pilih Berikutnya untuk pindah ke langkah berikutnya di wizard.

Langkah 3: Tambahkan versi produk awal

Halaman ini memandu Anda dengan menambahkan versi awal produk Anda. Produk Anda mungkin memiliki beberapa versi sepanjang siklus hidupnya, dan setiap versi diidentifikasi oleh AI SageMaker ARN yang unik.

1. Di bawah Nama Sumber Daya Amazon (ARNs):
 - a. Masukkan model atau algoritma Amazon SageMaker AI ARN.
 - Contoh paket model ARN: `arn:aws:sagemaker:<region>:<account-id>:model-package/<model-package-name>`
 - Untuk menemukan paket model ARN Anda, lihat Paket [model pasar saya](#).
 - Contoh algoritma ARN: `arn:aws:sagemaker:<region>:<account-id>:algorithm/<algorithm-name>`
 - Untuk menemukan ARN sumber daya algoritme Anda, lihat Algoritma [saya](#).
 - b. Masukkan peran akses IAM ARN.
 - Contoh IAM ARN: `arn:aws:iam::<account-id>:role/<role-name>`
2. Di bawah Informasi versi, masukkan nama Versi dan Catatan rilis. .
3. Di bawah Detail input Model, masukkan ringkasan input model dan berikan data input sampel untuk input pekerjaan real-time dan batch. Secara opsional, Anda dapat memberikan batasan input apa pun.
4. (Opsional) Di bawah parameter Input, berikan informasi rinci tentang setiap parameter input yang didukung oleh produk Anda. Anda dapat memberikan nama parameter, deskripsi, kendala, dan menentukan apakah parameter diperlukan atau opsional. Anda dapat memberikan hingga 24 parameter input.
5. (Opsional) Di bawah atribut Kustom, berikan parameter pemanggilan kustom yang didukung oleh produk Anda. Untuk setiap atribut, Anda dapat memberikan nama, deskripsi, batasan, dan menentukan apakah atribut diperlukan atau opsional.

6. Di bawah Detail keluaran Model, masukkan ringkasan output model dan berikan data keluaran sampel untuk output pekerjaan real-time dan batch. Secara opsional, Anda dapat memberikan batasan output apa pun.
7. (Opsional) Di bawah parameter Output, berikan informasi rinci tentang setiap parameter output yang didukung oleh produk Anda. Anda dapat memberikan nama parameter, deskripsi, kendala, dan menentukan apakah parameter diperlukan atau opsional. Anda dapat memberikan hingga 24 parameter output.
8. Di bawah Petunjuk penggunaan, berikan instruksi yang jelas untuk menggunakan model Anda secara efektif seperti praktik terbaik, cara menangani kasus tepi umum, atau saran pengoptimalan kinerja.
9. Di bawah repositori Git dan tautan notebook, berikan tautan ke buku catatan contoh dan repositori Git. Contoh notebook harus menyertakan cara memanggil model Anda. Repositori Git Anda harus menyertakan buku catatan, file data, dan alat pengembang lainnya.
10. Di bawah Jenis instans yang direkomendasikan, pilih jenis instans yang direkomendasikan untuk produk Anda.

Untuk paket model, Anda akan memilih jenis instans yang direkomendasikan untuk transformasi batch dan inferensi waktu nyata.

Untuk paket algoritme, Anda akan memilih jenis instans yang direkomendasikan untuk pekerjaan pelatihan.

Note

Jenis instans yang tersedia untuk dipilih terbatas pada yang didukung oleh model atau paket algoritme Anda. Jenis instans yang didukung ini ditentukan saat Anda pertama kali membuat sumber daya di Amazon SageMaker AI. Ini memastikan bahwa produk Anda hanya terkait dengan konfigurasi perangkat keras yang dapat menjalankan solusi pembelajaran mesin Anda secara efektif.

11. Pilih Berikutnya untuk pindah ke langkah berikutnya di wizard.

Langkah 4: Konfigurasi model harga

Saat mengonfigurasi model harga produk Anda, Anda dapat menawarkan produk Anda secara gratis atau menerapkan harga berbasis penggunaan. Model harga Anda tidak dapat diubah setelah Anda mempublikasikan produk.

1. Pilih model harga. Transformasi Batch dan produk pelatihan algoritma hanya dapat gratis atau dikenakan biaya untuk penggunaan per jam.
 - Jika Anda memilih untuk menawarkan produk Anda secara gratis, pilih Berikutnya dan lanjutkan wizard.
 - Jika Anda memilih harga penggunaan, lanjutkan langkah-langkah ini.
2. Jika Anda memilih untuk mengenakan biaya berdasarkan penggunaan, Anda dapat memasukkan biaya penggunaan. Anda dapat memilih untuk memasukkan harga yang berlaku untuk semua jenis instans atau memasukkan harga per jenis instans untuk harga yang lebih terperinci.
3. Pilih Ya, tawarkan uji coba gratis jika Anda ingin menawarkan uji coba gratis produk Anda.
4. Pilih Berikutnya untuk pindah ke langkah berikutnya di wizard.

Langkah 5: Konfigurasikan kebijakan pengembalian dana

Meskipun Anda tidak diharuskan menawarkan pengembalian uang, Anda harus mengajukan kebijakan pengembalian dana resmi. AWS Marketplace

1. Masukkan kebijakan pengembalian dana.
2. Pilih Berikutnya untuk pindah ke langkah berikutnya di wizard.

Langkah 6: Konfigurasikan EULA

Pada langkah ini, Anda akan memilih perjanjian hukum yang akan mengatur bagaimana pelanggan dapat menggunakan produk Anda. Anda dapat memilih AWS persyaratan kontrak standar atau mengunggah perjanjian lisensi pengguna akhir kustom Anda sendiri (EULA).

1. Pilih kontrak standar atau berikan perjanjian lisensi pengguna akhir kustom.
2. Pilih Berikutnya untuk pindah ke langkah berikutnya di wizard.

Langkah 7: Konfigurasikan allowlist

Sebelum mengirimkan produk Anda, Anda harus menentukan mana yang Akun AWS dapat mengaksesnya. Langkah opsional ini mengontrol visibilitas awal produk Anda, membatasi akses ke akun Anda sendiri dan setiap otorisasi khusus yang Akun AWS Anda tambahkan ke daftar yang diizinkan.

1. Masukkan yang Akun AWS IDs Anda inginkan untuk mengakses produk Anda.
2. Pilih Kirim untuk mengirimkan produk Anda.

Produk Anda akan memiliki status visibilitas terbatas dan hanya akan terlihat oleh Akun AWS yang membuat produk dan daftar izin lainnya. Akun AWS

Untuk informasi lebih lanjut tentang status, lihat [the section called “Status produk”](#).

Anda dapat melihat dan menguji daftar produk Anda saat berada dalam visibilitas terbatas. Ketika Anda siap untuk mengubah visibilitas produk Anda, lihat [the section called “”](#).

Mengelola produk pembelajaran mesin Anda

Di Portal AWS Marketplace Manajemen, pilih Minta perubahan untuk memodifikasi produk atau versi AWS Marketplace. Ketika Anda mengirimkan perubahan Anda, sistem memprosesnya. Waktu pemrosesan bervariasi dari menit ke hari, tergantung pada jenis modifikasi. Anda dapat memantau status perubahan Anda di Portal AWS Marketplace Manajemen.

Anda dapat memodifikasi produk ML AWS Marketplace melalui layanan mandiri dengan:

- [Memperbarui informasi produk](#)
- [Memperbarui visibilitas produk](#)
- [Memperbarui daftar yang diizinkan](#)
- [Mengelola versi produk](#)
- [Memperbarui harga produk](#)
- [Memperbarui kebijakan pengembalian dana Anda](#)
- [Memperbarui EULA Anda](#)
- [Menghapus produk](#)

Note

Selain melakukan perubahan melalui Portal AWS Marketplace Manajemen, Anda juga dapat melakukan perubahan menggunakan [AWS Marketplace Catalog API](#).

Memperbarui informasi produk

Setelah membuat produk machine learning (ML) Anda, Anda dapat memodifikasi informasi produk tertentu AWS Marketplace, seperti deskripsi, sorotan, judul, SKU, kategori, dan kata kunci.

1. Masuk ke akun penjual Anda di [Portal AWS Marketplace Manajemen](#).
2. Buka halaman Produk pembelajaran mesin dan pilih produk target Anda.
3. Pilih Minta perubahan dan pilih Perbarui informasi produk.
4. Perbarui bidang sesuai kebutuhan.

Note

Untuk spesifikasi logo, lihat [the section called “Persyaratan logo perusahaan dan produk”](#).

5. Pilih Kirim.

Anda dapat memantau permintaan Anda dari tab Permintaan pada halaman Produk pembelajaran mesin. Untuk informasi lebih lanjut tentang status, lihat [the section called “Status produk”](#).

Memperbarui visibilitas produk

1. Masuk ke akun penjual Anda di [Portal AWS Marketplace Manajemen](#).
2. Buka halaman produk pembelajaran mesin dan pilih produk Anda.
3. Pilih Minta perubahan, pilih Perbarui visibilitas produk, lalu pilih Publik atau Dibatasi.
4. Tinjau perubahan Anda dan pilih Kirim.

Anda dapat memantau permintaan Anda dari tab Permintaan pada halaman Produk pembelajaran mesin. Untuk informasi lebih lanjut tentang status, lihat [the section called “Status produk”](#).

Memperbarui daftar yang diizinkan

1. Masuk ke akun penjual Anda di [Portal AWS Marketplace Manajemen](#).
2. Buka halaman produk pembelajaran mesin dan pilih produk Anda.
3. Pilih Minta perubahan dan pilih Perbarui daftar yang diizinkan.
4. Ubah informasi yang perlu Anda ubah dan pilih Kirim. Untuk informasi selengkapnya, lihat [the section called “Langkah 7: Konfigurasi allowlist”](#).

Anda dapat memantau permintaan Anda dari tab Permintaan pada halaman Produk pembelajaran mesin. Untuk informasi lebih lanjut tentang status, lihat [the section called “Status produk”](#).

Mengelola versi produk

Sebagai penjual, Anda dapat mengelola versi produk Anda AWS Marketplace dengan memperbarui informasi versi yang ada, menambahkan versi baru, atau menghapus versi yang tidak lagi didukung. Setiap versi memiliki SageMaker AI ARN unik dan informasi terkait yang digunakan pembeli untuk mengevaluasi dan menyebarkan produk Anda.

Note

Sebelum menambahkan versi, buat ID produk dan tetapkan harga. Untuk informasi selengkapnya, lihat [the section called “Langkah 1: Buat daftar baru”](#).

Memperbarui informasi versi

Setelah membuat versi, Anda dapat memodifikasi informasi terkait seperti catatan rilis, petunjuk penggunaan, dan rekomendasi instans.

Note

Nama versi dan ARNs tidak dapat diubah. Perubahan ini memerlukan pembuatan versi baru.

1. Masuk ke akun penjual Anda di [Portal AWS Marketplace Manajemen](#).
2. Buka halaman produk pembelajaran mesin dan pilih produk Anda.
3. Pilih Minta perubahan dan pilih Perbarui informasi versi.
4. Pilih versi yang ingin Anda perbarui.
5. Pilih Edit versi.
6. Ubah bidang yang diperlukan dan pilih Berikutnya.
7. Masukkan informasi harga Anda dan pilih Kirim. Untuk informasi selengkapnya, lihat [the section called “Langkah 4: Konfigurasi model harga”](#).

Anda dapat memantau permintaan Anda dari tab Permintaan pada halaman Produk pembelajaran mesin. Untuk informasi lebih lanjut tentang status, lihat [the section called “Status produk”](#).

Menambahkan versi baru

Anda dapat menambahkan versi baru produk Anda untuk memperkenalkan fitur, pembaruan, atau peningkatan sambil mempertahankan akses ke versi sebelumnya.

1. Masuk ke akun penjual Anda di [Portal AWS Marketplace Manajemen](#).
2. Buka halaman produk pembelajaran mesin dan pilih produk Anda.
3. Pilih Versi dan pilih Tambahkan versi baru.
4. Masukkan informasi untuk versi baru mengikuti langkah-langkah di [the section called “Langkah 3: Tambahkan versi produk awal”](#).
5. Masukkan informasi harga Anda dan pilih Kirim. Untuk informasi selengkapnya, lihat [the section called “Langkah 4: Konfigurasi model harga”](#).

Ketika Anda telah berhasil menambahkan versi baru, pembeli menerima pemberitahuan email bahwa versi baru tersedia.

Membatasi versi

Ketika versi menjadi usang atau Anda ingin menghentikan ketersediaannya, Anda dapat membatasi akses pembeli ke versi itu sambil mempertahankan akses ke versi lain.

1. Masuk ke akun penjual Anda di [Portal AWS Marketplace Manajemen](#).
2. Buka halaman produk pembelajaran mesin dan pilih produk Anda.
3. Pilih Versi dan pilih Batasi versi.

Note

Anda harus selalu memiliki setidaknya satu versi yang tersedia.

4. Pilih Kirim.

Ketika Anda telah berhasil membatasi versi, pembeli menerima pemberitahuan email bahwa versi tersebut dibatasi.

Memperbarui harga produk

Anda dapat mengubah tarif dan periode uji coba gratis produk pembelajaran mesin Anda di AWS Marketplace, meskipun model penetapan harga itu sendiri tidak dapat diubah. Perhatikan bahwa untuk model berbayar, kenaikan harga berlaku setelah periode pemberitahuan 90 hari, pada hari pertama bulan berikutnya. Perubahan harga tambahan tidak dapat dilakukan selama periode pemberitahuan ini.

1. Masuk ke akun penjual Anda di [Portal AWS Marketplace Manajemen](#).
2. Buka halaman produk pembelajaran mesin dan pilih produk Anda.
3. Pilih Minta perubahan, pilih Perbarui penawaran umum, lalu pilih Edit informasi penawaran.
4. Ubah informasi yang perlu Anda ubah dan pilih Kirim.

Anda dapat memantau permintaan Anda dari tab Permintaan pada halaman Produk pembelajaran mesin. Untuk informasi lebih lanjut tentang status, lihat [the section called “Status produk”](#).

Memperbarui kebijakan pengembalian dana Anda

1. Masuk ke akun penjual Anda di [Portal AWS Marketplace Manajemen](#).
2. Buka halaman produk pembelajaran mesin dan pilih produk Anda.
3. Pilih Minta perubahan, pilih Perbarui penawaran umum, lalu pilih Perbarui kebijakan pengembalian dana.
4. Ubah informasi yang perlu Anda ubah dan pilih Kirim.

Anda dapat memantau permintaan Anda dari tab Permintaan pada halaman Produk pembelajaran mesin. Untuk informasi lebih lanjut tentang status, lihat [the section called “Status produk”](#).

Memperbarui EULA Anda

1. Masuk ke akun penjual Anda di [Portal AWS Marketplace Manajemen](#).
2. Buka halaman produk pembelajaran mesin dan pilih produk Anda.
3. Pilih Minta perubahan, pilih Perbarui penawaran umum, lalu pilih Perbarui EULA.
4. Ubah informasi yang perlu Anda ubah dan pilih Kirim.

Anda dapat memantau permintaan Anda dari tab Permintaan pada halaman Produk pembelajaran mesin. Untuk informasi lebih lanjut tentang status, lihat [the section called “Status produk”](#).

Menghapus produk

Anda dapat menghapus (matahari terbenam) produk Anda yang diterbitkan dari AWS Marketplace. Setelah dihapus, pelanggan baru tidak dapat berlangganan, tetapi Anda harus mendukung pelanggan yang sudah ada setidaknya selama 90 hari.

Berikut ini adalah kondisi untuk menghapus produk dari AWS Marketplace:

- Produk akan dihapus dari alat AWS Marketplace pencarian dan penemuan.
- Fungsionalitas berlangganan akan dinonaktifkan.
- Halaman detail produk tetap dapat diakses melalui URL langsung.
- Pelanggan saat ini mempertahankan akses sampai mereka membatalkan langganan mereka.
- AWS Marketplace memberi tahu pembeli saat ini tentang penghapusan.

Untuk menghapus produk pembelajaran mesin Anda:

1. Masuk ke akun penjual Anda di [Portal AWS Marketplace Manajemen](#).
2. Buka halaman produk pembelajaran mesin dan pilih produk Anda.
3. Pilih Minta perubahan, pilih Perbarui visibilitas produk, lalu pilih Dibatasi.
4. (Opsional) Masukkan ID produk pengganti.
5. Tinjau perubahan dan kemudian pilih Kirim.

Anda dapat memantau permintaan Anda dari tab Permintaan pada halaman Produk pembelajaran mesin. Untuk informasi lebih lanjut tentang status, lihat [the section called “Status produk”](#).

Setelah dihapus, produk muncul di daftar Produk saat ini di mana Anda hanya dapat mengunduh spreadsheet produk. Jika Anda memiliki pertanyaan tentang menghapus produk, hubungi tim [Operasi AWS Marketplace Penjual](#).

Membuat penawaran pribadi untuk produk pembelajaran mesin

Anda dapat bernegosiasi dan menawarkan penawaran pribadi langsung kepada pelanggan untuk produk pembelajaran mesin Anda. Untuk informasi lebih lanjut tentang penawaran pribadi, lihat [Mempersiapkan penawaran pribadi Anda](#).

Prasyarat:

- Anda harus memiliki daftar berbayar AWS Marketplace.
- Anda harus memiliki akses ke Portal AWS Marketplace Manajemen (AMMP).

Untuk membuat penawaran pribadi untuk produk pembelajaran mesin:

1. Masuk ke Portal AWS Marketplace Manajemen.
2. Pilih Penawaran, lalu pilih Buat penawaran pribadi
3. Pada halaman Buat penawaran pribadi, pilih produk yang ingin Anda buat penawaran pribadi. Anda hanya dapat membuat penawaran untuk produk yang tersedia.
4. Pada halaman Detail Penawaran:
 - a. Masukkan nama penawaran dan deskripsi.
 - b. Pilih opsi perpanjangan.
 - c. Tetapkan tanggal kedaluwarsa penawaran. Penawaran kedaluwarsa pada 23:59:59 UTC pada tanggal yang ditentukan.
5. Pilih Selanjutnya dua kali.
6. Pada halaman Konfigurasi harga dan durasi penawaran, tentukan:
 - Opsi harga

(Untuk informasi selengkapnya, lihat [the section called “Harga produk pembelajaran mesin”](#).)
 - Durasi kontrak
 - Tawarkan mata uang
 - Dimensi produk

Note

Untuk informasi selengkapnya tentang paket cicilan, lihat [Paket cicilan penawaran pribadi](#).

7. Pilih Berikutnya.
8. Pada halaman Tambah pembeli, masukkan Akun AWS IDs untuk pembeli Anda. Lalu pilih Selanjutnya.

⚠ Important

Untuk akun tertaut untuk mendapatkan keuntungan dari penawaran pribadi:

- Sertakan ID AWS akun pembayar.
- Akun pembayar harus menerima ketentuan per jam dari penawaran pribadi terlebih dahulu.
- Setelah akun pembayar menerima, akun yang ditautkan kemudian dapat menerima penawaran pribadi.

9. Pada halaman Konfigurasi persyaratan hukum dan tawarkan dokumen, tambahkan persyaratan khusus apa pun, lalu pilih Berikutnya.

ℹ Note

Anda dapat menambahkan hingga lima file (ketentuan hukum, pernyataan kerja, tagihan bahan, lembar harga, atau addendum). Sistem menggabungkan ini menjadi satu dokumen.

10. Pada halaman Tinjau dan buat, verifikasi detail penawaran dan pilih Buat penawaran.
11. Setelah penawaran muncul di halaman Kelola penawaran pribadi, buka menu Tindakan, pilih Salin URL penawaran, dan kirimkan email ke pembeli.

ℹ Note

Penawaran mungkin membutuhkan waktu untuk dipublikasikan. Anda dapat mengedit penawaran di halaman Kelola penawaran pribadi hingga pembeli menerimanya.

Persyaratan dan praktik terbaik untuk membuat produk pembelajaran mesin

Adalah penting bahwa pembeli Anda merasa mudah untuk menguji paket model dan produk algoritma Anda. Bagian berikut menjelaskan praktik terbaik untuk produk ML. Untuk ringkasan lengkap persyaratan dan rekomendasi, lihat [Ringkasan persyaratan dan rekomendasi untuk daftar produk ML](#).

Note

AWS Marketplace Perwakilan dapat menghubungi Anda untuk membantu Anda memenuhi persyaratan ini jika produk yang Anda publikasikan tidak memenuhi persyaratan tersebut.

Topik

- [Praktik terbaik umum untuk produk ML](#)
- [Persyaratan untuk informasi penggunaan](#)
- [Persyaratan untuk input dan output](#)
- [Persyaratan untuk notebook Jupyter](#)
- [Ringkasan persyaratan dan rekomendasi untuk daftar produk ML](#)

Praktik terbaik umum untuk produk ML

Berikan informasi berikut untuk produk pembelajaran mesin Anda:

- Untuk deskripsi produk, sertakan yang berikut ini:
 - Apa yang dilakukan model Anda
 - Siapa target pelanggan
 - Apa kasus penggunaan yang paling penting
 - Bagaimana model Anda dilatih atau jumlah data yang digunakan
 - Apa metrik kinerja dan data validasi yang digunakan
 - Jika medis, apakah model Anda untuk penggunaan diagnostik atau tidak
- Secara default, produk pembelajaran mesin dikonfigurasi untuk memiliki visibilitas publik. Namun, Anda dapat membuat produk dengan visibilitas terbatas. Untuk informasi selengkapnya, lihat [the section called “Langkah 7: Konfigurasi allowlist”](#).
- (Opsional) Untuk produk berbayar, tawarkan uji coba gratis selama 14-30 hari bagi pelanggan untuk mencoba produk Anda. Untuk informasi selengkapnya, lihat [Harga produk pembelajaran mesin untuk AWS Marketplace](#).

Persyaratan untuk informasi penggunaan

Informasi penggunaan yang jelas yang menggambarkan input dan output yang diharapkan dari produk Anda (dengan contoh) sangat penting untuk mendorong pengalaman pembeli yang positif.

Dengan setiap versi baru sumber daya Anda yang Anda tambahkan ke daftar produk Anda, Anda harus memberikan informasi penggunaan.

Untuk mengedit informasi penggunaan yang ada untuk versi tertentu, lihat [the section called “Memperbarui informasi versi”](#).

Persyaratan untuk input dan output

Penjelasan yang jelas tentang parameter input yang didukung dan parameter output yang dikembalikan dengan contoh penting untuk membantu pembeli Anda memahami dan menggunakan produk Anda. Pemahaman ini membantu pembeli Anda untuk melakukan transformasi yang diperlukan pada data input untuk mendapatkan hasil inferensi terbaik.

Anda akan diminta untuk hal berikut saat menambahkan sumber daya Amazon SageMaker AI Anda ke daftar produk Anda.

Input dan output inferensi

Untuk input inferensi, berikan deskripsi data input yang diharapkan produk Anda untuk pekerjaan endpoint real-time dan batch transform. Sertakan cuplikan kode untuk setiap pra-pemrosesan data yang diperlukan. Sertakan batasan, jika berlaku. Berikan sampel masukan yang dihosting di [GitHub](#).

Untuk output inferensi, berikan deskripsi data keluaran yang dikembalikan produk Anda untuk pekerjaan endpoint real-time dan batch transform. Sertakan batasan, jika berlaku. Berikan sampel keluaran yang dihosting di [GitHub](#).

Untuk sampel, berikan file input yang berfungsi dengan produk Anda. Jika model Anda melakukan klasifikasi multiclass, berikan setidaknya satu file input sampel untuk setiap kelas.

Input pelatihan

Di bagian Informasi untuk melatih model, berikan format data input dan cuplikan kode untuk setiap pra-pemrosesan data yang diperlukan. Sertakan deskripsi nilai dan batasan, jika berlaku. Berikan sampel masukan yang dihosting di [GitHub](#).

Jelaskan fitur opsional dan wajib yang dapat disediakan oleh pembeli, dan tentukan apakah mode PIPE input didukung. Jika [pelatihan terdistribusi](#) (pelatihan dengan lebih dari 1 instance CPU/GPU) didukung, tentukan ini. Untuk penyetelan, daftarkan hyperparameters yang direkomendasikan.

Persyaratan untuk notebook Jupyter

Saat menambahkan sumber daya SageMaker AI Anda ke daftar produk Anda, berikan tautan ke contoh notebook Jupyter yang dihosting [GitHub](#) yang menunjukkan alur kerja lengkap tanpa meminta pembeli untuk mengunggah atau menemukan data apa pun.

Gunakan AWS SDK for Python (Boto). Notebook sampel yang dikembangkan dengan baik memudahkan pembeli untuk mencoba dan menggunakan daftar Anda.

Untuk produk paket model, notebook sampel Anda mendemonstrasikan persiapan data input, pembuatan titik akhir untuk inferensi waktu nyata, dan kinerja pekerjaan transformasi batch. Untuk informasi selengkapnya, lihat [daftar Model Package dan Contoh notebook](#) di GitHub. Untuk contoh buku catatan, lihat [auto_insurance](#). Notebook berfungsi secara keseluruhan Wilayah AWS, tanpa memasukkan parameter apa pun dan tanpa pembeli yang perlu mencari data sampel.

Note

Contoh notebook Jupyter yang belum berkembang yang tidak menunjukkan beberapa kemungkinan input dan langkah pra-pemrosesan data mungkin menyulitkan pembeli untuk sepenuhnya memahami proposisi nilai produk Anda.

Untuk produk algoritme, notebook sampel menunjukkan pelatihan lengkap, penyetelan, pembuatan model, pembuatan titik akhir untuk inferensi waktu nyata, dan kinerja pekerjaan transformasi batch. Untuk informasi selengkapnya, lihat [Daftar algoritma dan Contoh buku catatan](#) di GitHub. [Untuk contoh buku catatan, lihat amazon_demo_product dan automl on](#). GitHub Notebook sampel ini bekerja di semua Wilayah tanpa memasukkan parameter apa pun dan tanpa pembeli yang perlu mencari data sampel.

Note

Kurangnya contoh data pelatihan dapat mencegah pembeli Anda menjalankan notebook Jupyter dengan sukses. Notebook sampel yang kurang berkembang dapat mencegah pembeli Anda menggunakan produk Anda dan menghambat adopsi.

Ringkasan persyaratan dan rekomendasi untuk daftar produk ML

Tabel berikut memberikan ringkasan persyaratan dan rekomendasi untuk halaman daftar produk pembelajaran mesin.

Detail	Untuk daftar paket model	Untuk daftar algoritma
Product descriptions		
Jelaskan secara rinci apa yang dilakukan produk untuk jenis konten yang didukung (misalnya, "mendeteksi X dalam gambar").	Wajib	Wajib
Berikan informasi yang menarik dan membedakan tentang produk (hindari kata sifat seperti klaim "terbaik" atau tidak berdasar).	Disarankan	Disarankan
Sebutkan kasus penggunaan terpenting untuk produk ini.	Wajib	Wajib
Jelaskan data (sumber dan ukuran) yang dilatih dan buat daftar batasan yang diketahui.	Wajib	Tidak berlaku
Jelaskan kerangka inti tempat model dibangun.	Disarankan	Disarankan
Ringkas metrik kinerja model pada data validasi (misalnya, "Akurasi persen XX.YY yang dibenchmark menggunakan dataset Z").	Wajib	Tidak berlaku
Meringkas latensi model dan/ atau metrik throughput pada	Wajib	Tidak berlaku

Detail	Untuk daftar paket model	Untuk daftar algoritma
jenis instans yang direkomen- dasikan.		
Jelaskan kategori algoritma. Misalnya, “Algoritma regresi hutan keputusan ini didasarka- n pada ansambel pengklasi- fikasi terstruktur pohon yang dibangun menggunakan teknik umum agregasi bootstrap dan pilihan fitur acak.”	Tidak berlaku	Wajib
Usage information		
Untuk inferensi, berikan deskripsi format input yang diharapkan untuk pekerjaan endpoint real-time dan batch transform. Sertakan batasan, jika berlaku. Lihat Persyaratan untuk input dan output .	Wajib	Wajib
Untuk inferensi, berikan sampel input untuk pekerjaan endpoint real-time dan batch transform. Sampel harus di- host di GitHub. Lihat Persyarat- an untuk input dan output .	Wajib	Wajib
Untuk inferensi, berikan nama dan deskripsi setiap parameter input. Berikan rincian tentang keterbatasannya dan tentukan apakah diperlukan atau opsional.	Disarankan	Disarankan

Detail	Untuk daftar paket model	Untuk daftar algoritma
Untuk inferensi, berikan detail tentang data keluaran yang dikembalikan produk Anda untuk pekerjaan endpoint real-time dan batch transform. Sertakan batasan apa pun, jika berlaku. Lihat Persyaratan untuk input dan output .	Wajib	Wajib
Untuk inferensi, berikan sampel keluaran untuk titik akhir real-time dan pekerjaan transformasi batch. Sampel harus di-host di GitHub. Lihat Persyaratan untuk input dan output .	Wajib	Wajib
Untuk inferensi, berikan contoh penggunaan pekerjaan endpoint atau batch transform . Sertakan contoh kode menggunakan perintah AWS Command Line Interface (AWS CLI) atau menggunakan AWS SDK.	Wajib	Wajib
Untuk inferensi, berikan nama dan deskripsi setiap parameter output. Tentukan apakah selalu dikembalikan.	Disarankan	Disarankan

Detail	Untuk daftar paket model	Untuk daftar algoritma
Untuk pelatihan, berikan detail tentang informasi yang diperlukan untuk melatih model seperti baris data minimum yang diperlukan. Lihat Persyaratan untuk input dan output .	Tidak berlaku	Wajib
Untuk pelatihan, berikan sampel masukan yang dihosting di GitHub. Lihat Persyaratan untuk input dan output .	Tidak berlaku	Wajib
Untuk pelatihan, berikan contoh melakukan pekerjaan pelatihan. Jelaskan hiperparameter yang didukung, rentangnya, dan dampak keseluruhannya. Tentukan apakah algoritme mendukung penyetelan hiperparameter, pelatihan terdistribusi, atau instance GPU. Sertakan contoh kode seperti AWS CLI perintah atau menggunakan AWS SDK, misalnya.	Tidak berlaku	Wajib
Sediakan notebook Jupyter yang dihosting untuk GitHub menunjukkan penggunaan lengkap produk Anda. Lihat Persyaratan untuk notebook Jupyter .	Wajib	Wajib

Detail	Untuk daftar paket model	Untuk daftar algoritma
Memberikan informasi teknis terkait penggunaan produk, termasuk manual pengguna dan data sampel.	Disarankan	Disarankan

Memecahkan masalah dengan produk pembelajaran mesin

Bagian ini memberikan bantuan untuk beberapa kesalahan umum yang mungkin Anda temui selama proses penerbitan untuk produk pembelajaran mesin Anda. Jika masalah Anda tidak tercantum, hubungi tim [Operasi AWS Marketplace Penjual](#).

Umum: Saya mendapatkan kesalahan 400 ketika saya menambahkan Nama Sumber Daya Amazon (ARN) dari paket model atau algoritme saya di Portal Manajemen AWS Marketplace

Penyebab umum

Saat membuat produk pembelajaran mesin Anda di SageMaker AI, Anda tidak memilih untuk mempublikasikan produk Anda AWS Marketplace.

Resolusi

Jika Anda menggunakan konsol Amazon SageMaker AI untuk membuat sumber daya Anda, Anda harus memilih Ya di halaman akhir proses untuk Publikasikan paket model ini di AWS Marketplace atau Ya untuk Publikasikan algoritme ini di AWS Marketplace. Anda tidak dapat memilih Tidak dan kemudian mempublikasikannya. Memilih Ya tidak mempublikasikan paket model atau algoritma. Namun, ini memvalidasi paket model atau sumber daya algoritme Anda saat dibuat, yang diperlukan untuk digunakan di AWS Marketplace.

Jika Anda menggunakan AWS SDK untuk [membuat paket model](#) atau [membuat algoritme](#), pastikan parameter `CertifyForMarketplace` disetel ke `true`.

Setelah Anda membuat ulang paket model atau sumber daya algoritme bersertifikat dan tervalidasi, tambahkan ARN baru di file. Portal Manajemen AWS Marketplace

Umum: Saya mendapatkan kesalahan 404 ketika saya menambahkan ARN paket model atau algoritme saya di Portal Manajemen AWS Marketplace

Penyebab umum

Kesalahan ini dapat terjadi karena beberapa alasan:

- ARN mungkin tidak valid.
- Paket model atau sumber daya algoritme tidak dibuat Akun AWS sama dengan akun penjual.
- Pengguna atau peran yang Anda gunakan untuk penerbitan tidak memiliki izin IAM yang benar untuk mengakses paket model atau sumber daya algoritme.

Resolusi

1. Periksa ARN untuk memastikannya adalah ARN yang benar dan dalam format yang diharapkan:

Untuk paket model, ARNs seharusnya terlihat mirip dengan `arn:aws:sagemaker:us-east-2:000123456789:model-package/my-model-package-name`.

Untuk algoritma, ARNs seharusnya terlihat mirip `arn:aws:sagemaker:us-east-2:000123456789:algorithm/my-algorithm` dengan.

2. Pastikan bahwa semua sumber daya dan aset untuk penerbitan ada di akun penjual tempat Anda menerbitkan.
3. Pastikan bahwa pengguna atau peran Anda memiliki izin berikut:

Untuk paket model, tindakan `sagemaker:DescribeModelPackage` pada sumber daya paket model harus diizinkan.

Untuk algoritme, tindakan `sagemaker:DescribeAlgorithm` pada sumber daya algoritme harus diizinkan.

Amazon SageMaker AI: Saya mendapatkan pesan kegagalan “Kesalahan klien: Akses ditolak untuk registri” saat saya membuat paket model atau sumber daya algoritme

Penyebab umum

Kesalahan ini dapat terjadi ketika gambar yang digunakan untuk membuat paket model atau algoritma disimpan dalam repositori [Amazon ECR](#) milik orang lain. Akun AWS Paket model atau validasi algoritma tidak mendukung gambar lintas akun.

Resolusi

Salin gambar ke repositori Amazon ECR yang dimiliki oleh Akun AWS yang Anda gunakan untuk mempublikasikan. Kemudian, lanjutkan dengan membuat sumber daya menggunakan lokasi gambar baru.

Amazon SageMaker AI: Saya mendapatkan “Tidak Dimulai” dan “Kesalahan klien: Tidak ada pemindaian yang dijadwalkan...” pesan kegagalan saat saya membuat paket model atau sumber daya algoritme

Penyebab umum

Kesalahan ini dapat terjadi ketika SageMaker AI gagal memulai pemindaian gambar kontainer Docker Anda yang disimpan di Amazon ECR.

Resolusi

Jika ini terjadi, buka [konsol Amazon ECR](#), temukan repositori tempat gambar Anda diunggah, pilih gambar, lalu pilih Pindai.

Produk berbasis SaaS di AWS Marketplace

Dengan produk perangkat lunak sebagai layanan (SaaS), Anda dapat menyebarkan perangkat lunak yang dihosting di AWS infrastruktur dan memberi AWS Marketplace pembeli akses ke perangkat lunak di lingkungan Anda. AWS Anda bertanggung jawab untuk mengelola akses pelanggan, pembuatan akun, penyediaan sumber daya, dan manajemen akun dalam perangkat lunak Anda. Setelah membuat produk SaaS awal, Anda dapat mengirimkan permintaan perubahan dan mengonfigurasi propertinya AWS Marketplace, termasuk deskripsi produk, ketersediaan menurut negara, harga, dan sebagainya. Anda juga mengonfigurasi langganan SaaS dan integrasi kontrak Anda. Topik berikut memberi Anda informasi yang Anda butuhkan untuk memulai.

Untuk informasi tentang mengintegrasikan Anda APIs dengan Amazon API Gateway, lihat [Menjual API Gateway APIs melalui AWS Marketplace](#) Panduan Pengembang Amazon API Gateway.

Untuk bantuan dengan produk SaaS Anda, hubungi tim [Operasi AWS Marketplace Penjual](#).

Topik

- [Memulai dengan produk SaaS di AWS Marketplace](#)
- [Merencanakan produk SaaS Anda](#)
- [Pedoman produk SaaS untuk AWS Marketplace](#)
- [Harga produk SaaS di AWS Marketplace](#)
- [Membuat penawaran uji coba gratis SaaS di AWS Marketplace](#)
- [Pelanggan orientasi ke produk SaaS Anda melalui AWS Marketplace](#)
- [Notifikasi Amazon SNS untuk produk SaaS](#)
- [Mengakses Layanan AWS Marketplace Pengukuran dan Hak APIs](#)
- [Pelaporan untuk produk SaaS di AWS Marketplace](#)
- [Contoh kode untuk integrasi produk SaaS](#)
- [Mengirimkan produk Anda melalui VPC Amazon menggunakan AWS PrivateLink](#)

Memulai dengan produk SaaS di AWS Marketplace

Sebagai AWS Marketplace penjual, Anda dapat menambahkan perangkat lunak Anda sebagai produk layanan (SaaS) ke. AWS Marketplace Ini termasuk mengintegrasikan produk SaaS Anda

dengan operasi API yang AWS Marketplace sesuai, berdasarkan model penagihan produk SaaS. Topik ini memberikan gambaran umum tentang proses pembuatan dan konfigurasi produk SaaS, dimulai dengan prasyarat yang diperlukan yang Anda butuhkan untuk memulai.

Prasyarat

Sebelum memulai, Anda harus menyelesaikan prasyarat berikut:

1. Akses dan gunakan [Portal Manajemen AWS Marketplace](#). Ini adalah alat yang Anda gunakan untuk mendaftar sebagai penjual dan mengelola produk yang Anda jual AWS Marketplace. Untuk informasi selengkapnya, lihat [Portal Manajemen AWS Marketplace](#).
2. Daftar sebagai penjual, dan kirimkan informasi pajak dan perbankan Anda. Untuk informasi selengkapnya, lihat [Mendaftar sebagai AWS Marketplace penjual](#).
3. Rencanakan bagaimana Anda akan membuat dan mengintegrasikan produk SaaS Anda. AWS Marketplace Untuk informasi selengkapnya, lihat [Merencanakan produk SaaS Anda](#).

Langkah selanjutnya

Setelah Anda menyelesaikan prasyarat yang diperlukan, Anda dapat membuat dan mengonfigurasi produk SaaS Anda.

Topik

- [Siklus hidup produk SaaS di AWS Marketplace](#)
- [Membuat produk SaaS di AWS Marketplace](#)
- [Membuat halaman produk SaaS awal di AWS Marketplace](#)
- [Mengkonfigurasi pengaturan produk SaaS di AWS Marketplace](#)
- [Mengintegrasikan produk langganan SaaS Anda dengan AWS Marketplace](#)
- [Mengintegrasikan produk kontrak SaaS Anda dengan AWS Marketplace](#)
- [Mengintegrasikan kontrak pay-as-you-go SaaS Anda dengan produk dengan AWS Marketplace](#)
- [Menerapkan integrasi SaaS tanpa server](#)

Siklus hidup produk SaaS di AWS Marketplace

Saat Anda membuat produk perangkat lunak sebagai layanan (SaaS) AWS Marketplace, awalnya diterbitkan dengan visibilitas terbatas sehingga hanya akun Anda yang dapat mengaksesnya. Ketika

Anda siap, Anda dapat mempublikasikannya ke AWS Marketplace katalog untuk memungkinkan pembeli berlangganan dan membeli produk Anda. Topik berikut memberikan informasi tentang siklus hidup produk SaaS. Untuk informasi selengkapnya tentang membuat produk SaaS, lihat, [Membuat produk SaaS di AWS Marketplace](#)

Pada halaman produk SaaS, Anda dapat melihat daftar produk Anda. Bergantung pada tahapannya dalam siklus hidup produk, produk akan memiliki salah satu status berikut:

- **Pementasan** — Produk tidak lengkap yang masih Anda tambahkan informasinya. Pada Simpan pertama dan keluar dari pengalaman swalayan, permintaan perubahan yang berhasil membuat produk yang tidak dipublikasikan dengan informasi dari langkah-langkah lengkap yang Anda kirimkan. Dari status ini, Anda dapat terus menambahkan informasi ke produk atau mengubah detail yang sudah dikirimkan melalui permintaan perubahan.
- **Terbatas** — Sebuah produk selesai setelah diserahkan ke sistem dan melewati semua validasi dalam sistem. Kemudian produk dirilis ke status Terbatas. Pada titik ini, produk memiliki halaman detail yang hanya dapat diakses oleh akun Anda dan siapa pun yang Anda izinkan. Anda dapat menguji produk Anda melalui halaman detail. Untuk informasi atau bantuan lebih lanjut, hubungi tim [Operasi AWS Marketplace Penjual](#).
- **Publik** — Saat Anda siap mempublikasikan produk sehingga pembeli dapat melihat dan berlangganan produk, Anda menggunakan permintaan perubahan visibilitas Perbarui. Permintaan ini memulai alur kerja untuk tim Operasi AWS Marketplace Penjual untuk meninjau dan mengaudit produk Anda terhadap AWS kebijakan. Setelah produk disetujui dan permintaan perubahan diproses, produk dipindahkan dari status Terbatas ke Publik. Untuk informasi tentang AWS pedoman, lihat pedoman [produk SaaS](#).
- **Dibatasi** - Jika Anda ingin menghentikan pengguna baru berlangganan produk Anda, Anda dapat membatasi produk dengan menggunakan permintaan perubahan visibilitas Perbarui. Status Terbatas berarti bahwa pengguna yang ada dapat terus menggunakan produk. Namun, produk tidak akan lagi terlihat oleh publik atau tersedia untuk pengguna baru.

Anda dapat memperbarui produk Anda di status Pementasan, Terbatas, dan Publik. Untuk informasi selengkapnya, lihat [Memperbarui informasi produk](#).

Membuat produk SaaS di AWS Marketplace

Sebagai AWS Marketplace penjual, Anda dapat menambahkan perangkat lunak Anda sebagai produk layanan (SaaS) ke AWS Marketplace. Ini termasuk membuat produk SaaS Anda dan mengintegrasikannya dengan operasi AWS Marketplace API yang sesuai, berdasarkan model

penagihan Anda. Untuk menjual perangkat lunak sebagai produk SaaS di AWS Marketplace, Anda mengikuti langkah-langkah luas ini:

- Buat produk SaaS di AWS Marketplace
- Integrasikan [langganan](#), [kontrak](#), atau [kontrak SaaS dengan pay-as-you-go produk dengan](#) AWS Marketplace
- Uji [langganan](#), [kontrak](#), atau [kontrak dengan](#) integrasi pay-as-you-go produk dengan AWS Marketplace.
- Kirimkan produk Anda untuk diluncurkan.

Prosedur berikut menunjukkan kepada Anda cara membuat produk SaaS di AWS Marketplace

Buat produk SaaS

Untuk membuat produk SaaS

1. Putuskan untuk membuat daftar produk SaaS

Miliki produk SaaS yang ingin Anda jual. AWS Marketplace Tinjau dan pahami bagaimana caranya [Merencanakan produk SaaS Anda](#).

2. Tentukan harga dan jenis penawaran

Ada tiga jenis penawaran untuk produk SaaS: langganan, kontrak, dan kontrak dengan. pay-as-you-go Pilihan jenis penawaran Anda memengaruhi cara Anda mengintegrasikan produk SaaS Anda. AWS Marketplace Untuk informasi selengkapnya, lihat [Rencanakan harga Anda](#).

3. Kumpulkan aset

Kumpulkan aset yang dibutuhkan untuk mengirimkan produk Anda. Aset meliputi:

- URL logo produk — URL Amazon S3 yang dapat diakses publik yang berisi gambar logo yang jelas untuk produk yang Anda berikan.
- URL Perjanjian Lisensi Pengguna Akhir (EULA) — Produk Anda harus memiliki EULA yang tersedia sebagai file PDF. Anda harus memberikan tautan ke bucket Amazon S3 tempat pelanggan dapat meninjau EULA di halaman produk Anda. AWS Marketplace
- URL pendaftaran produk — Ini adalah URL tempat pembeli dialihkan setelah berhasil berlangganan produk Anda. AWS Marketplace

- Metadata tentang produk Anda — Anda menyediakan metadata di wizard pembuatan produk. Portal Manajemen AWS Marketplace
 - Informasi Support untuk produk Anda — Informasi ini mencakup alamat email dan URLs saluran dukungan produk Anda.
4. Kirim produk Anda untuk integrasi

Gunakan akun penjual Anda dan Portal AWS Marketplace Manajemen untuk [Membuat halaman produk SaaS awal di AWS Marketplace](#). AWS Marketplace akan mempublikasikan produk Anda sebagai produk terbatas, yang berarti bahwa itu hanya tersedia untuk digunakan untuk integrasi dan pengujian. Kode produk Anda dan topik Amazon Simple Notification Service (SNS) akan tersedia untuk Anda di halaman ikhtisar produk.

 Note

Produk Anda harus tetap pada harga yang lebih murah sehingga Anda dan tim Operasi AWS Marketplace Penjual dapat menguji produk Anda tanpa menimbulkan biaya besar. Kami akan meminta harga aktual produk saat Anda meminta visibilitas publik untuk produk Anda.

5. Integrasikan dengan AWS Marketplace

Produk Anda harus mendukung orientasi pelanggan dan menggunakan produk Anda, termasuk memvalidasi langganan mereka sebelum memberi mereka akses, dan, dalam beberapa kasus, pengukuran untuk penggunaannya. Bagaimana Anda berintegrasi dengan AWS Marketplace tergantung pada jenis penawaran yang Anda gunakan untuk produk Anda. Untuk informasi selengkapnya tentang integrasi, berdasarkan jenis penawaran, lihat topik berikut:

- [Integrasi berlangganan](#)
- [Integrasi kontrak](#)
- [Kontrak dengan pay-as-you-go integrasi](#)

Langkah terakhir untuk mengintegrasikan produk Anda AWS Marketplace adalah mengujinya untuk memastikan bahwa integrasi berfungsi dengan baik.

6. Pengujian produk dan pembatalan kontrak

Setelah Anda menyelesaikan proses integrasi, kami sarankan Anda berlangganan produk Anda sendiri untuk mengevaluasi dan mengkonfirmasi pengalaman pelanggan. Membuat langganan uji memungkinkan Anda untuk:

- Tinjau informasi produk
- Periksa opsi pembelian yang tersedia
- Proses pembayaran uji
- Verifikasi tautan ke situs web produk Anda sendiri

 Important

Batalkan langganan pengujian Anda sebelum mengubah visibilitas produk Anda dari terbatas ke publik. Jika Anda memerlukan bantuan, hubungi [tim Marketplace Seller Operations](#) dengan mengirimkan tiket dukungan.

7. Kirim produk Anda untuk diluncurkan

Setelah memverifikasi integrasi dan siap untuk produk ditayangkan, pilih Perbarui visibilitas. Tim Operasi AWS Marketplace Penjual akan meninjau produk Anda dan memperbarui harga sebelum visibilitas dapat diperbarui ke Publik.

 Note

AWS Marketplace Operasi Penjual menggunakan proses manual untuk memverifikasi dan memperbarui produk SaaS. Proses ini memakan waktu 7-10 hari kerja untuk memperbarui visibilitas ke publik, dan lebih lama jika tim menemukan kesalahan. Untuk informasi lebih lanjut tentang pengaturan waktu, lihat [Pengaturan waktu dan harapan](#) dalam panduan ini.

Membuat halaman produk SaaS awal di AWS Marketplace

Anda dapat menggunakan metadata aplikasi perangkat lunak sebagai layanan (SaaS) Anda untuk membuat halaman produk SaaS awal di katalog, menggunakan AWS Marketplace Portal Manajemen AWS Marketplace Anda juga dapat menambahkan informasi produk, detail penyebaran

produk, dan detail penawaran umum. Secara opsional, Anda dapat menambahkan akun ke daftar yang diizinkan untuk menguji produk. Untuk informasi lebih lanjut lihat prosedur berikut.

Untuk membuat halaman produk SaaS awal

1. Masuk ke [Portal Manajemen AWS Marketplace](#).
2. Pilih Buat produk SaaS, lalu pilih produk SaaS.
3. Hasilkan ID dan kode produk SaaS. Anda juga dapat menambahkan tag opsional untuk mendukung otorisasi berbasis tag.

 Note

Untuk informasi tentang otorisasi berbasis tag, lihat [Mengontrol akses ke AWS sumber daya menggunakan tag](#) di AWS Identity and Access Management Panduan Pengguna.

4. Gunakan pengalaman swalayan untuk membuat AWS Marketplace daftar. Tambahkan informasi produk, detail penyebaran produk, dan detail penawaran umum. Secara opsional, Anda juga dapat menambahkan akun ke daftar yang diizinkan untuk menguji produk.

 Note

Jika Anda perlu mengakhiri sesi sebelum menyelesaikan langkah-langkahnya, pilih opsi Simpan dan keluar untuk menyimpan pilihan Anda saat ini ke area pementasan. Opsi ini membuat permintaan untuk memvalidasi informasi yang Anda berikan. Saat permintaan Anda sedang divalidasi, Anda tidak dapat mengedit produk. Jika permintaan Anda berhasil, Anda dapat terus membuat produk Anda dengan memilih Lanjutkan pembuatan produk.

Jika permintaan Anda tidak berhasil, itu karena kesalahan validasi, yang terlihat pada log permintaan produk. Pilih permintaan untuk melihat kesalahan, dan pilih Salin ke baru di bawah Tindakan untuk memperbaiki kesalahan dan mengirim ulang permintaan. Untuk memperbarui langkah sebelumnya, buka halaman detail produk dan kirimkan permintaan perubahan.

 Note

Harga Anda akan default ke \$0,001 per dimensi selama pengujian. Harga ini memungkinkan Anda untuk menguji produk Anda dalam keadaan Terbatas tanpa

menimbulkan tagihan besar. Anda akan memberikan harga aktual Anda saat membuat produk Anda menjadi publik.

5. Pilih Kirim. Kemudian, AWS Marketplace memvalidasi informasi. Jika validasi berhasil, lepaskan AWS Marketplace produk dalam status Terbatas. Setelah validasi berhasil, Anda dapat melihat pratinjau, mengintegrasikan, dan menguji produk Anda.

Note

Saat validasi sedang berlangsung, Anda tidak dapat mengedit produk. Ketika produk Anda awalnya diterbitkan, itu hanya dapat diakses oleh yang Akun AWS digunakan untuk membuat produk dan akun pengujian tim Operasi AWS Marketplace Penjual. Jika Anda melihat produk dari halaman produk SaaS, Anda dapat memilih Lihat AWS Marketplace untuk melihat detail produk karena akan muncul AWS Marketplace untuk pembeli. Daftar detail ini tidak terlihat oleh pengguna AWS Marketplace .

Mengkonfigurasi pengaturan produk SaaS di AWS Marketplace

Setelah Anda [membuat produk perangkat lunak sebagai layanan \(SaaS\)](#) AWS Marketplace, Anda dapat memodifikasi banyak pengaturan produk. Bagian berikut menunjukkan cara mengirimkan permintaan perubahan dan mengubah pengaturan produk, seperti memperbarui detail harga, visibilitas produk, dan pengaturan lainnya.

Topik

- [Mengelola permintaan perubahan](#)
- [Perbarui informasi produk](#)
- [Perbarui detail arsitektur](#)
- [Perbarui daftar yang diizinkan Akun AWS IDs](#)
- [Perbarui visibilitas produk](#)
- [Perbarui persyaratan harga](#)
- [Tambahkan dimensi harga](#)
- [Perbarui dimensi harga](#)
- [Batasi dimensi harga](#)
- [Tentukan bagaimana pembeli akan mengakses produk Anda](#)

- [Konfigurasi Peluncuran Cepat](#)
- [Perbarui ketersediaan menurut negara](#)
- [Perbarui kebijakan pengembalian dana suatu produk](#)
- [Perbarui perjanjian lisensi pengguna akhir \(EULA\)](#)

Mengelola permintaan perubahan

Dalam [daftar layanan mandiri](#), Anda menggunakan permintaan perubahan untuk membuat perubahan pada produk Anda. Permintaan Anda saat ini dapat ditemukan Portal Manajemen AWS Marketplace di [tab Permintaan](#). Anda dapat membuat permintaan baru melalui daftar dropdown Permintaan perubahan yang terletak di bawah bilah navigasi.

Untuk membuat permintaan perubahan untuk produk SaaS

1. Buka [Portal Manajemen AWS Marketplace](#), dan masuk ke akun penjual Anda.
2. Dari tab Produk, pilih SaaS dari daftar dropdown.
3. Setelah permintaan diajukan, itu mulai diproses. Permintaan perubahan melewati status berikut: Sedang ditinjau, Mempersiapkan perubahan, dan Menerapkan perubahan.
4. Ketika pemrosesan permintaan selesai, statusnya berubah menjadi salah satu nilai berikut:
 - Berhasil - Status ini menunjukkan bahwa perubahan yang Anda minta telah diproses dan perubahan tercermin dalam sistem.
 - Gagal - Status ini menunjukkan bahwa ada yang tidak beres dengan permintaan dan perubahan tidak diproses. Jika status Gagal, Anda dapat memilih permintaan untuk menemukan Kode Kesalahan yang memberikan rekomendasi tentang cara memperbaiki masalah. Anda dapat memecahkan masalah kesalahan dan membuat permintaan baru untuk perubahan. Untuk membuat proses lebih cepat, Anda dapat menggunakan fungsi Salin ke permintaan baru yang menyalin rincian permintaan Gagal. Anda dapat membuat perubahan yang diperlukan dan mengirimkan kembali permintaan.

Permintaan perubahan yang dimulai dengan pembaruan akan memuat detail proyek saat ini. Kemudian, Anda dapat membuat pembaruan, yang menampilkan detail yang ada. Tambah dan batasi pasangan permintaan khusus untuk pembaruan yang disediakan setelah setiap permintaan berhasil (setelah Anda memilih Simpan dan keluar dan Kirim tindakan dalam pengalaman layanan mandiri). Ini berarti pelanggan yang ada dapat terus menggunakan produk sampai langganan atau kontrak

mereka berakhir. Namun, tidak ada pelanggan baru yang dapat ditambahkan ke produk yang berada dalam status Terbatas.

Perbarui informasi produk

Setelah Anda membuat produk Anda, Anda mungkin ingin mengubah informasi yang terkait dengannya AWS Marketplace.

1. Buka [Portal Manajemen AWS Marketplace](#), dan masuk ke akun penjual Anda.
2. Dari halaman [Produk SaaS](#), pada tab produk SaaS, pilih produk yang ingin Anda modifikasi.
3. Dari daftar tarik-turun Permintaan perubahan, pilih Perbarui informasi produk.
4. Perbarui salah satu bidang berikut yang ingin Anda ubah:
 - Judul produk
 - SKU
 - Deskripsi singkat
 - Deskripsi panjang
 - URL logo produk
 - Sorotan
 - Kategori produk
 - Kata kunci
 - URL video produk
 - Sumber Daya
 - Informasi Support

Note

Untuk detail tentang format logo, lihat [Persyaratan logo perusahaan dan produk](#).

5. Untuk memperbarui informasi produk, pilih Kirim.
6. Verifikasi bahwa permintaan muncul di tab Permintaan dengan status Dalam tinjauan. Anda mungkin perlu me-refresh halaman untuk melihat permintaan baru Anda.

Perbarui detail arsitektur

Untuk menerima penunjukan khusus yang digunakan produk Anda AWS, perbarui detail arsitektur produk Anda di Portal AWS Marketplace Manajemen (AMMP) dengan memilih pola hosting dan mengunggah diagram arsitektur. Untuk daftar pola hosting yang AWS mempertimbangkan untuk [Pedoman](#) digunakan AWS, lihat.

Untuk memperbarui detail arsitektur

1. Masuk ke [Portal Manajemen AWS Marketplace](#).
2. Pada menu Produk, pilih SaaS.
3. Di tabel produk SaaS, pilih produk Anda.
4. Untuk perubahan Permintaan, pilih Perbarui detail arsitektur.
5. Pilih pola hosting. Jika Anda memilih pola hosting yang AWS Marketplace mempertimbangkan untuk digunakan AWS, selesaikan dua langkah tambahan berikut. Untuk informasi lebih lanjut tentang pola yang dianggap digunakan AWS, lihat [Pedoman](#), sebelumnya dalam panduan ini.
 - a. Jika diminta, di bagian Diagram arsitektur, pilih Pilih file untuk mengunggah diagram arsitektur Anda dalam format PNG atau JPG. Untuk informasi lebih lanjut tentang diagram, lihat [Membuat diagram arsitektur](#)
 - b. Jika diminta, di bidang Aplikasi, pilih di mana aplikasi Anda berjalan. Untuk informasi lebih lanjut, lihat [Control plane vs. bidang aplikasi](#).
6. Pilih Perbarui detail architecture.

Setelah permintaan selesai, status Permintaan berubah menjadi Berhasil. Untuk memeriksa status permintaan, pilih produk Anda di tabel produk SaaS dan pilih tab Permintaan log.

Untuk melihat hasil penilaian Anda, pilih produk Anda di tabel produk SaaS dan pilih tab Detail arsitektur.

Perbarui daftar yang diizinkan Akun AWS IDs

Anda dapat mengubah daftar Akun AWS IDs yang dapat melihat produk Anda dalam keadaan terbatas.

1. Buka [Portal Manajemen AWS Marketplace](#), dan masuk ke akun penjual Anda.
2. Dari halaman [produk SaaS](#), pada tab produk SaaS, pilih produk yang ingin Anda modifikasi.

3. Dari daftar tarik-turun Permintaan perubahan, pilih Perbarui daftar yang diizinkan. Daftar menunjukkan daftar Akun AWS IDs yang saat ini diizinkan.
4. Di Akun AWS bidang Allowlisted, masukkan Akun AWS IDs dan pisahkan menggunakan koma.
5. Untuk memperbarui daftar yang diizinkan Akun AWS IDs, pilih Kirim.

Perbarui visibilitas produk

Untuk mengubah pembeli mana yang dapat melihat pengalaman Peluncuran Cepat Anda AWS Marketplace, Anda dapat menggunakan visibilitas Pembaruan.

1. Buka [Portal Manajemen AWS Marketplace](#), dan masuk ke akun penjual Anda.
2. Dari halaman [produk SaaS](#), pilih produk yang ingin Anda modifikasi.
3. Dari menu tarik-turun Permintaan perubahan, pilih Perbarui visibilitas.

Note

Anda dapat meminta agar produk dipindahkan dari status Terbatas ke status Publik dengan menggunakan permintaan perubahan ini. Namun, permintaan perubahan harus melalui proses persetujuan tim Operasi AWS Marketplace Penjual untuk dipindahkan ke Publik.

4. Ketika Anda mempublikasikan ke publik, Anda akan memberikan harga aktual untuk produk Anda. Harga ini akan diterapkan setelah iklan Anda disetujui untuk visibilitas publik.
5. Untuk mengirimkan permintaan Anda untuk ditinjau, pilih Kirim.
6. Verifikasi bahwa tab Permintaan menampilkan status Permintaan sebagai Dalam peninjauan. Ketika permintaan selesai, status menjadi Berhasil.

Perbarui persyaratan harga

Untuk mengubah harga per dimensi pada produk SaaS Anda, gunakan Perbarui ketentuan harga.

Note

Kenaikan harga untuk dimensi apa pun mengakibatkan opsi pembaruan harga tidak tersedia setidaknya selama 90 hari ke depan. Jika memperbarui penurunan harga dan kenaikan, perbarui penurunan harga terlebih dahulu.

1. Buka [Portal Manajemen AWS Marketplace](#) dan masuk ke akun penjual Anda.
2. Dari halaman [Produk SaaS](#), pada tab produk SaaS, pilih produk yang ingin Anda modifikasi.
3. Dari daftar tarik-turun Permintaan perubahan, pilih Perbarui penawaran umum, lalu pilih Perbarui persyaratan harga.
4. Harga saat ini sudah diisi sebelumnya di bidang. Anda dapat menghapus harga saat ini, dan kemudian menambahkan harga baru Anda.
5. Untuk mengirimkan permintaan Anda untuk ditinjau, pilih Kirim.
6. Verifikasi bahwa tab Permintaan menampilkan status Permintaan sebagai Dalam peninjauan. Ketika permintaan selesai, status akan diperbarui ke Berhasil atau Gagal.

Tambahkan dimensi harga

Anda dapat menambahkan dimensi yang ingin Anda gunakan untuk mengisi daya produk Anda. Dimensi adalah unit ukuran dasar yang dikenakan pembeli Anda saat menggunakan produk Anda.

Note

Untuk memperbarui nama atau deskripsi dimensi harga yang ada, lihat [the section called "Perbarui dimensi harga"](#).

1. Buka [Portal Manajemen AWS Marketplace](#), dan masuk ke akun penjual Anda.
2. Dari tab [Produk SaaS](#), pilih produk yang ingin Anda modifikasi.
3. Dari menu tarik-turun Permintaan perubahan, pilih Perbarui dimensi harga, lalu Tambahkan dimensi harga.
4. Berikan pengenal API dimensi, nama tampilan, dan deskripsi untuk menambahkan dimensi baru ke produk Anda, lalu pilih Berikutnya.

Note

Pengenal dan nama API harus unik di semua dimensi. Anda tidak dapat mengubah pengenal dan unit API setelah dimensi dibuat.

5. Tentukan harga untuk setiap dimensi yang telah Anda tambahkan, lalu pilih Berikutnya untuk meninjau perubahan Anda.

Note

Anda hanya dapat menambahkan dimensi untuk model harga yang dipilih untuk produk Anda (misalnya, kontrak, penggunaan, atau kontrak dengan konsumsi). Untuk produk terbatas, harga untuk dimensi yang baru ditambahkan ditetapkan ke \$0,01. Anda dapat memperbarui harga saat produk siap untuk visibilitas publik.

6. Pilih Kirim untuk mengirimkan permintaan Anda untuk ditinjau.
7. Di tab Permintaan, verifikasi bahwa status permintaan sedang dalam peninjauan. Ketika permintaan selesai, status berubah menjadi Succeeded.

Perbarui dimensi harga

Anda dapat memperbarui dimensi yang ingin Anda gunakan untuk mengisi daya produk Anda. Dimensi adalah unit ukuran dasar yang dikenakan pembeli Anda saat menggunakan produk Anda.

1. Buka [Portal Manajemen AWS Marketplace](#), dan masuk ke akun penjual Anda.
2. Dari tab [Produk SaaS](#), pilih produk yang ingin Anda modifikasi.
3. Dari menu tarik-turun Permintaan perubahan, pilih Perbarui dimensi harga, lalu Perbarui informasi dimensi.
4. Temukan dimensi yang ingin Anda perbarui, lalu pilih nama atau deskripsi.
5. Berikan nama atau deskripsi baru, lalu pilih tanda centang untuk mengonfirmasi pembaruan Anda. Nama dimensi harus unik.
6. Pilih Kirim untuk mengirimkan permintaan Anda untuk ditinjau.
7. Di tab Permintaan, verifikasi bahwa status permintaan sedang dalam peninjauan. Ketika permintaan selesai, status akan berubah menjadi Succeeded.

Batasi dimensi harga

Anda dapat membatasi dimensi yang saat ini tercantum dalam produk. Permintaan ini menghapus dimensi yang dipilih dari produk.

1. Buka [Portal Manajemen AWS Marketplace](#), dan masuk ke akun penjual Anda.
2. Dari tab [Produk SaaS](#), pilih produk yang ingin Anda modifikasi.

3. Dari menu tarik-turun Permintaan perubahan, pilih Perbarui dimensi harga, lalu Batasi dimensi harga.
4. Untuk produk terbatas dan publik, Anda akan diminta untuk menghubungi tim Operasi AWS Marketplace Penjual menggunakan tombol [Hubungi Kami](#). Menggunakan formulir, berikan detail untuk dimensi yang ingin Anda hapus dari daftar produk Anda.

 Note

Operasi mungkin tidak selalu dapat membatasi dimensi harga.

Tentukan bagaimana pembeli akan mengakses produk Anda

Anda dapat memilih salah satu opsi berikut untuk bagaimana pelanggan dapat mengakses produk Anda:

- [the section called “Perbarui opsi pemenuhan URL SaaS”](#)— Pelanggan menggunakan URL untuk situs tempat mereka dialihkan setelah berlangganan produk Anda. AWS Marketplace
- [the section called “Konfigurasi Peluncuran Cepat”](#)— Pelanggan menggunakan proses yang disederhanakan untuk mengkonfigurasi dan meluncurkan produk Anda. Anda dapat menyelesaikan konfigurasi ini untuk produk yang ada dengan visibilitas Terbatas atau Publik.

Perbarui opsi pemenuhan URL SaaS

Untuk memperbarui URL yang digunakan untuk memenuhi produk SaaS Anda, gunakan tab Opsi Pemenuhan Perbarui.

1. Buka [Portal Manajemen AWS Marketplace](#), dan masuk ke akun penjual Anda.
2. Dari halaman [Produk SaaS](#), pada tab produk SaaS, pilih produk yang ingin Anda modifikasi.
3. Dari daftar tarik-turun Permintaan perubahan, pilih opsi Perbarui pemenuhan, lalu pilih Edit URL pemenuhan default.
4. Di bidang URL Pemenuhan, masukkan URL baru untuk opsi pemenuhan produk SaaS.
5. Untuk mengirimkan permintaan Anda untuk ditinjau, pilih Kirim.
6. Verifikasi bahwa tab Permintaan menampilkan status Permintaan sebagai Dalam peninjauan. Ketika permintaan selesai, status akan diperbarui ke Berhasil atau Gagal.

Konfigurasi Peluncuran Cepat

Produk SaaS yang terdaftar AWS Marketplace sering membutuhkan AWS sumber daya untuk digunakan di akun pembeli berlangganan (misalnya, peran IAM). Quick Launch memungkinkan Anda memberi pembeli panduan, step-by-step instruksi, dan penyebaran sumber daya menggunakan AWS CloudFormation templat. Pembeli menggunakan CloudFormation template untuk mengkonfigurasi dan meluncurkan produk. Untuk mempelajari lebih lanjut tentang proses konfigurasi Quick Launch, lihat lab [Aktifkan SaaS Quick Launch](#).

Untuk mengonfigurasi pengalaman Peluncuran Cepat yang dapat digunakan pelanggan untuk meluncurkan produk SaaS Anda, gunakan tab Opsi Pemenuhan.

1. Dari halaman [Produk SaaS](#), pada tab produk SaaS, pilih produk yang ingin Anda modifikasi.

Note

Untuk mengonfigurasi pengalaman Peluncuran Cepat, produk harus memiliki visibilitas Terbatas atau Publik.

2. Pada halaman detail produk, pilih tab Opsi pemenuhan.
3. Untuk Quick Launch, pilih tombol Activate and configure.
4. Untuk detail login Akun, berikan URL untuk situs Anda tempat pembeli dapat masuk atau membuat akun. URL ini membuka tab baru dalam pengalaman pembeli. Pembeli kemudian masuk atau membuat akun dan kembali AWS Marketplace untuk meluncurkan template.
5. Buat AWS CloudFormation template.

Tip

Ikuti AWS Well-Architected Framework saat membuat template AWS CloudFormation Anda yang menyebarkan sumber daya ke akun pembeli. AWS Untuk informasi dan sumber daya selengkapnya, [AWS Well-Architected](#) dan baca [AWS Well-Architected Framework](#).

Untuk AWS CloudFormation template, pilih tombol Add AWS CloudFormation template dan berikan informasi berikut:

- Judul — Berikan nama CloudFormation penyebaran Anda.

- Deskripsi - Berikan deskripsi template.
- Nama tumpukan - Berikan nama untuk tumpukan. Nama ini adalah nama tumpukan untuk pembeli di CloudFormation.
- CloudFormation URL template - Menyediakan Amazon Simple Storage Service akan meninjau template ini.

 Note

Untuk menyederhanakan proses peluncuran bagi pelanggan Anda, kami sarankan meminimalkan jumlah template yang terkait dengan proses konfigurasi Anda. Idealnya, Anda menginginkan satu template yang menyebarkan sumber daya yang dibutuhkan untuk menggunakan produk. Untuk pertanyaan yang terkait dengan CloudFormation template Anda, hubungi mitra pengembangan AWS Marketplace bisnis Anda atau tim [Operasi AWS Marketplace Penjual](#).

- Izin IAM yang diperlukan - Berikan izin yang diperlukan untuk menyebarkan template. CloudFormation Jika Anda ingin berbagi parameter penerapan, yang disimpan sebagai rahasia [AWS Secrets Manager](#) untuk pembeli, kebijakan Anda harus menyertakan tindakan berikut:
 - `secretsManager:ListSecrets`
 - `secretsManager:DescribeSecret`
 - `secretsManager:ReplicateSecretToRegions`
 - `secretsManager:GetSecretValue`

 Note

Jika produk Anda memerlukan parameter CloudFormation penerapan yang disediakan penjual (misalnya, kunci API dan [eksternal IDs](#)), gunakan `PutDeploymentParameter` operasi untuk berbagi parameter dengan pelanggan Anda. Untuk informasi selengkapnya, lihat [PutDeploymentParameter](#) di Referensi API Layanan AWS Marketplace Penerapan.

6. (Opsional) Untuk petunjuk konfigurasi Manual, berikan instruksi untuk pembeli yang ingin mengonfigurasi produk Anda secara manual. Pertimbangkan untuk menyertakan tautan ke panduan orientasi dan dokumentasi produk Anda.

7. Untuk detail Peluncuran, berikan URL tempat pembeli akan mengakses produk setelah CloudFormation tumpukan digunakan.
8. (Opsional) Untuk akun yang Diizinkan untuk Peluncuran Cepat, berikan daftar terpisah koma Akun AWS yang dapat melihat pengalaman Peluncuran Cepat dengan visibilitas terbatas.
9. Pilih tombol Kirim. Pengalaman Quick Launch akan memiliki visibilitas terbatas, artinya hanya akan terlihat oleh akun Anda dan akun yang diizinkan. Dengan Visibilitas terbatas, Anda dapat menguji konfigurasi menggunakan halaman Konfigurasi dan peluncuran setelah berlangganan produk dan memilih tombol Siapkan akun Anda.
 - Untuk melihat pratinjau halaman dari perspektif pembeli, pilih Pratinjau. Perubahan konfigurasi Anda akan tercermin pada pratinjau saat kedua halaman terbuka.
10. Jika sudah siap, Anda dapat mempublikasikan pengalaman Quick Launch di AWS Marketplace katalog. Gunakan tombol Update Quick Launch visibility pada tab Fulfillment options pada halaman detail produk.

Saat Anda mengubah visibilitas ke Publik, tim Operasi AWS Marketplace Penjual akan meninjau konfigurasi, melakukan pengujian pembeli, dan mempublikasikan pengalaman.

 Note

Jika Anda memerlukan dukungan saat mengaktifkan pengalaman Peluncuran Cepat, hubungi tim [Operasi AWS Marketplace Penjual](#).

Perbarui ketersediaan menurut negara

Anda dapat menentukan negara di mana produk Anda dapat ditawarkan.

1. Buka [Portal Manajemen AWS Marketplace](#), dan masuk ke akun penjual Anda.
2. Dari tab [Produk SaaS](#), pilih produk yang ingin Anda modifikasi.
3. Dari menu tarik-turun Permintaan perubahan, pilih Perbarui penawaran umum, lalu pilih Perbarui ketersediaan menurut negara.
4. Pilih salah satu opsi berikut:
 - Semua negara - Tersedia di semua negara yang didukung.
 - Semua negara dengan pengecualian — Tersedia di semua negara yang didukung kecuali di negara tertentu.

- Hanya negara yang diizinkan — Daftar spesifik negara tempat produk tersedia.
5. Pilih Kirim untuk mengirimkan permintaan Anda untuk ditinjau.
 6. Di tab Permintaan, verifikasi bahwa status permintaan sedang dalam peninjauan. Ketika permintaan selesai, status akan berubah menjadi Succeeded.

Perbarui kebijakan pengembalian dana suatu produk

Anda dapat memperbarui kebijakan pengembalian dana untuk produk Anda dengan menggunakan kebijakan pengembalian dana Perbarui.

1. Buka [Portal Manajemen AWS Marketplace](#), dan masuk ke akun penjual Anda.
2. Dari halaman [Produk SaaS](#), pada tab produk SaaS, pilih produk yang ingin Anda modifikasi.
3. Dari daftar tarik-turun Permintaan perubahan, pilih Perbarui penawaran umum, lalu pilih Perbarui kebijakan pengembalian dana.
4. Rincian kebijakan pengembalian dana saat ini disediakan di kotak teks. Tinjau dan modifikasi detail yang Anda inginkan. Mengirimkan permintaan menimpa kebijakan pengembalian dana saat ini.
5. Untuk mengirimkan permintaan Anda untuk ditinjau, pilih Kirim.
6. Verifikasi bahwa tab Permintaan menampilkan status Permintaan sebagai Dalam peninjauan. Ketika permintaan selesai, status akan diperbarui ke Berhasil atau Gagal.

Perbarui perjanjian lisensi pengguna akhir (EULA)

Anda dapat memperbarui EULA Anda untuk pengguna baru yang berlangganan produk Anda.

1. Buka [Portal Manajemen AWS Marketplace](#), dan masuk ke akun penjual Anda.
2. Dari tab [Produk SaaS](#), pilih produk yang ingin Anda modifikasi.
3. Dari menu tarik-turun Permintaan perubahan, pilih Perbarui penawaran umum dan kemudian Perbarui EULA.
4. Anda dapat memilih [Kontrak Standar untuk AWS Marketplace \(SCMP\)](#) atau mengirimkan EULA khusus. Untuk EULA kustom, Anda harus memberikan URL Amazon Simple Storage Service (Amazon S3) untuk kontrak. Bucket Amazon S3 Anda harus dapat diakses publik.
5. Pilih Kirim untuk mengirimkan permintaan Anda untuk ditinjau.

6. Di tab Permintaan, verifikasi bahwa status permintaan sedang dalam peninjauan. Ketika permintaan selesai, status akan berubah menjadi Succeeded.

Mengintegrasikan produk langganan SaaS Anda dengan AWS Marketplace

Mengintegrasikan produk Anda dengan AWS Marketplace adalah salah satu langkah. [Membuat produk SaaS di AWS Marketplace](#) Untuk mengintegrasikan perangkat lunak Anda sebagai produk berlangganan layanan (SaaS) AWS Marketplace, Anda harus menulis kode dan menunjukkan bahwa ia dapat merespons dengan sukses beberapa skenario pelanggan. Bagian berikut menunjukkan cara mengintegrasikan model langganan SaaS Anda. AWS Marketplace

Note

Sebelum Anda mulai, pastikan Anda telah memilih model harga yang tepat untuk produk software-as-a-service (SaaS) Anda. AWS Marketplace Untuk informasi selengkapnya, lihat [Merencanakan produk SaaS Anda](#).

Topik

- [Skenario: Layanan Anda memvalidasi pelanggan baru](#)
- [Skenario: Penggunaan meter](#)
- [Skenario: Pantau perubahan langganan pengguna](#)
- [Skenario: Verifikasi langganan pelanggan](#)
- [Menguji integrasi produk langganan SaaS Anda](#)

Skenario: Layanan Anda memvalidasi pelanggan baru

Ketika pelanggan berlangganan produk Anda, mereka diarahkan ke URL pendaftaran Anda yang merupakan permintaan HTTP POST dengan token sementara `amzn-marketplace-token`. Tanggapi permintaan ini dengan cara berikut:

1. Tukarkan token dengan `CustomerIdentifierCustomerAWSAccountId`, dan `ProductCode` dengan memanggil operasi [ResolveCustomer](#) API di AWS Marketplace Metering Service.
2. Pertahankan `CustomerIdentifier`, `CustomerAWSAccountID`, dan `ProductCode` dalam sistem Anda untuk panggilan future. Anda harus menyimpan apakah pelanggan memiliki

langganan yang valid, bersama dengan informasi apa pun yang Anda butuhkan tentang pelanggan.

3. Sebagai tanggapan atas permintaan tersebut, Anda harus menunjukkan pengalaman penggunaan pertama pengguna Anda (sebagaimana berlaku untuk layanan Anda).

Skenario: Penggunaan meter

Ketika pelanggan mulai menggunakan layanan Anda, Anda harus mengirim catatan pengukuran setiap jam. Untuk detail tentang cara meteran, lihat [Mengkonfigurasi pengukuran untuk penggunaan dengan langganan SaaS](#).

Kami menyarankan Anda menggunakan AWS CloudTrail untuk memantau aktivitas untuk memastikan bahwa informasi penagihan sedang dikirim ke AWS. Ingatlah hal berikut saat mengirim catatan pengukuran:

- Permintaan pengukuran tidak digandakan pada jam tersebut.
- Catatan yang dikirim setiap jam bersifat kumulatif.
- Kami sangat menyarankan sebagai praktik terbaik bahwa, bahkan jika tidak ada catatan dalam satu jam terakhir, Anda mengirim catatan pengukuran setiap jam, dengan penggunaan 0.

Skenario: Pantau perubahan langganan pengguna

Siapkan antrian Amazon Simple Queue Service (Amazon SQS), dan berlangganan topik Amazon SNS produk Anda. Informasi topik SNS Anda disertakan dalam pesan email yang Anda terima dari tim Operasi AWS Marketplace Penjual saat Anda membuat produk. Untuk informasi selengkapnya, lihat [Membuat produk SaaS di AWS Marketplace](#). Dengan berlangganan topik SNS Anda, Anda menerima pemberitahuan tentang perubahan langganan pelanggan, termasuk menyediakan atau mencabut akses untuk pelanggan tertentu.

Note

Topik Amazon SNS Nama Sumber Daya Amazon (ARN) terlihat seperti. `arn:aws:sns:us-east-1:<account id>:aws-mp-subscription-notification-<product code>`

Notifikasi yang harus Anda tanggapi adalah:

- `subscribe-success`— Pelanggan berlangganan, dan Anda dapat berhasil mengukur ID pelanggan mereka.
- `unsubscribe-pending`— Pelanggan sedang dalam proses berhenti berlangganan. Anda harus mengirim catatan pengukuran terakhir.
- `unsubscribe-success`— Pelanggan telah berhenti berlangganan. Catatan pengukuran untuk pelanggan tidak akan lagi diterima. Ikuti praktik Anda untuk mematikan sumber daya pelanggan, mengikuti kebijakan retensi Anda.
- `subscribe-fail`— Langganan pelanggan gagal. Anda tidak boleh meteran terhadap ID pelanggan mereka atau membuat sumber daya atas nama pelanggan.

Skenario: Verifikasi langganan pelanggan

Sebelum membuat sumber daya atas nama pelanggan, verifikasi bahwa pelanggan harus memiliki akses ke produk Anda. Simpan status terbaru pelanggan dari notifikasi yang Anda terima melalui Amazon SQS untuk mengetahui apakah pelanggan memiliki akses.

Menguji integrasi produk langganan SaaS Anda

Setelah Anda mengintegrasikan produk langganan SaaS Anda AWS Marketplace, Anda harus melakukan pengujian mendalam untuk memastikan bahwa integrasi berhasil. Prosedur berikut menguraikan langkah-langkah untuk memverifikasi integrasi produk Anda.

Note

Gunakan akun Anda sendiri untuk berlangganan produk Anda dan uji apakah integrasi berhasil. Harga dapat dikurangi sementara sehingga Anda dapat menguji alur pembelian tanpa menimbulkan biaya tinggi di akun tersebut. Untuk informasi selengkapnya tentang mengurangi harga sementara atau mengizinkan akun pengujian tambahan mengakses produk Anda, hubungi tim [Operasi AWS Marketplace Penjual](#). Setelah produk Anda diluncurkan, layanan harus terus menanggapi skenario ini untuk pelanggan baru.

1. Gunakan akun yang diizinkan untuk menguji pengalaman pelanggan dengan berlangganan produk Anda.
2. Setelah berlangganan akun yang diizinkan, pastikan akun dialihkan ke URL pendaftaran, dan pengalihan adalah permintaan POST yang menyertakan token sementara. Pastikan aplikasi

Anda tetap menggunakan ID pelanggan untuk panggilan future. Ini menguji bagian dari [Skenario: Layanan Anda memvalidasi pelanggan baru](#).

3. Setelah memverifikasi akun pengujian pada langkah sebelumnya, masukkan akun ke dalam aplikasi Anda. Misalnya, Anda dapat meminta pelanggan uji mengisi formulir untuk membuat pengguna baru. Atau, berikan mereka langkah selanjutnya untuk mendapatkan akses ke aplikasi SaaS Anda. Ini menguji bagian dari [Skenario: Layanan Anda memvalidasi pelanggan baru](#).
4. Setelah pelanggan pengujian onboard, buat permintaan yang akan mengirim catatan pengukuran AWS untuk tujuan penagihan dengan menggunakan operasi `BatchMeterUsage` API di `AWS Marketplace Metering Service` Tes ini [Skenario: Penggunaan meter](#).
5. Uji perubahan langganan. Skenario yang mungkin termasuk berhenti berlangganan, langganan yang berhasil, dan langganan yang gagal. Tes ini [Skenario: Pantau perubahan langganan pengguna](#).
6. Verifikasi langganan yang berhasil. Setelah Anda menerima notifikasi Amazon SNS untuk akun pengujian Anda dengan pesan berlangganan yang berhasil, pengukuran dapat dimulai. Catatan yang dikirim ke `AWS Marketplace Metering Service` sebelum Anda menerima notifikasi Amazon SNS tidak diukur. Tes ini [Skenario: Verifikasi langganan pelanggan](#).

Note

Untuk mencegah masalah penagihan, kami sangat menyarankan untuk menunggu pemberitahuan ini secara terprogram sebelum meluncurkan sumber daya atas nama pelanggan Anda.

7. Setelah Anda menyelesaikan semua persyaratan integrasi dan menguji solusinya, beri tahu tim Operasi AWS Marketplace Penjual. Mereka akan menjalankan serangkaian tes akhir pada solusi dengan memverifikasi bahwa Anda telah berhasil mengirim catatan terukur dengan operasi `BatchMeterUsage` API.

Setelah integrasi dan pengujian selesai, Anda dapat melakukan tinjauan akhir dan mencantumkan produk Anda di depan umum AWS Marketplace. Untuk informasi selengkapnya, lihat [Membuat produk SaaS di AWS Marketplace](#).

Mengintegrasikan produk kontrak SaaS Anda dengan AWS Marketplace

Mengintegrasikan perangkat lunak Anda sebagai produk kontrak layanan (SaaS) AWS Marketplace dengan adalah satu langkah. [Membuat produk SaaS di AWS Marketplace](#) Untuk mengintegrasikan

produk kontrak perangkat lunak Anda sebagai layanan (SaaS) AWS Marketplace, Anda harus menulis kode dan menunjukkan bahwa ia dapat merespons dengan sukses beberapa skenario pelanggan. Bagian berikut menjelaskan skenario ini, menjelaskan cara menanggapi, dan memberikan gambaran umum tentang pengujian integrasi Anda.

Note

Sebelum Anda mulai, pastikan Anda telah memilih model harga yang tepat untuk perangkat lunak Anda sebagai produk layanan (SaaS). AWS Marketplace Untuk informasi selengkapnya, lihat [Merencanakan produk SaaS Anda](#).

Topik

- [Skenario: Layanan Anda memvalidasi pelanggan baru](#)
- [Skenario: Layanan Anda menangani permintaan pelanggan](#)
- [Skenario: Pantau perubahan langganan pengguna](#)
- [Menguji integrasi produk kontrak SaaS Anda](#)

Skenario: Layanan Anda memvalidasi pelanggan baru

Ketika pelanggan berlangganan produk Anda, mereka diarahkan ke URL pendaftaran Anda, yang merupakan permintaan HTTP POST dengan token sementara `amzn-marketplace-token`. Tanggapi permintaan ini dengan cara berikut:

1. Tukarkan token dengan `CustomerIdentifierCustomerAWSAccountId`, dan `ProductCode` dengan memanggil operasi [ResolveCustomer](#) API di AWS Marketplace Metering Service.
2. Verifikasi langganan dan kuantitas (jika ada) yang dapat diakses pelanggan dengan memanggil operasi [GetEntitlements](#) API di AWS Marketplace Entitlement Service.
3. Pertahankan `CustomerIdentifier`, `CustomerAWSAccountId`, dan `ProductCode` dalam sistem Anda untuk panggilan future. Simpan apakah pelanggan memiliki langganan yang valid, bersama dengan informasi apa pun yang Anda butuhkan tentang pelanggan.
4. Sebagai tanggapan atas permintaan tersebut, Anda harus menunjukkan pengalaman penggunaan pertama pengguna Anda (sebagaimana berlaku untuk layanan Anda).

Skenario: Layanan Anda menangani permintaan pelanggan

Ketika pelanggan membuat permintaan ke layanan Anda, Anda harus menanggapi skenario berikut dengan tindakan atau pesan yang sesuai:

- Mereka tidak memiliki ID pelanggan di sistem Anda. Ini berarti mereka belum berlangganan. Anda harus memberi tahu pengguna cara berlangganan.
- Mereka memiliki ID pelanggan, dan operasi `GetEntitlements` API mengembalikan hak yang sesuai. Dalam skenario ini, Anda harus memenuhi permintaan.
- Mereka memang memiliki ID pelanggan, tetapi operasi `GetEntitlements` API tidak mengembalikan hak, atau jumlah yang tidak cukup untuk memenuhi permintaan. Dalam skenario ini, Anda harus menentukan cara menangani akses dan mengelola pengalaman mereka.

Skenario: Pantau perubahan langganan pengguna

Siapkan antrian Amazon Simple Queue Service (Amazon SQS), dan berlangganan topik Amazon SNS produk Anda. Informasi topik SNS Anda disertakan dalam pesan email yang Anda terima dari Tim AWS Marketplace Operasi saat Anda membuat produk. Untuk informasi selengkapnya, lihat [Membuat produk SaaS di AWS Marketplace](#). Dengan berlangganan topik SNS Anda, Anda menerima pemberitahuan tentang perubahan hak pelanggan, termasuk menyediakan atau mencabut akses untuk pelanggan tertentu.

Note

Topik SNS Nama Sumber Daya Amazon (ARN) terlihat seperti. `arn:aws:sns:us-east-1:<account id>:aws-mp-entitlement-notification-<product code>`

Satu-satunya pemberitahuan yang harus Anda tanggapi adalah:

- `entitlement-updated`— Hak pelanggan telah berubah, dan Anda harus memanggil operasi `GetEntitlements` API untuk melihat status baru. Perbarui toko pelanggan Anda, dan, jika berlaku (misalnya, kontrak pelanggan telah berakhir), ikuti praktik Anda untuk mematikan sumber daya pelanggan, mengikuti kebijakan retensi Anda.

Note

Untuk informasi tambahan, lihat [Memeriksa hak menggunakan AWS Marketplace Entitlement Service](#).

Menguji integrasi produk kontrak SaaS Anda

Setelah Anda mengintegrasikan produk kontrak SaaS Anda AWS Marketplace, Anda harus melakukan pengujian mendalam untuk memastikan bahwa integrasi berhasil. Prosedur berikut menguraikan langkah-langkah untuk memverifikasi integrasi produk Anda.

Note

Gunakan akun Anda sendiri untuk berlangganan produk Anda dan uji apakah integrasi berhasil. Harga dapat dikurangi sementara sehingga Anda dapat menguji alur pembelian tanpa menimbulkan biaya tinggi di akun tersebut. Untuk informasi lebih lanjut tentang pengurangan sementara harga atau mengizinkan akun uji tambahan untuk mengakses produk Anda, [hubungi kami](#).

Setelah produk Anda diluncurkan, layanan harus terus menanggapi skenario ini untuk pelanggan baru.

1. Gunakan akun yang diizinkan untuk menguji pengalaman pelanggan dengan mendapatkan kontrak untuk produk Anda.
2. Setelah akun memiliki kontrak, pastikan bahwa akun dialihkan ke URL pendaftaran, dan bahwa pengalihan adalah permintaan POST yang menyertakan token sementara. Pastikan aplikasi Anda mempertahankan ID pelanggan untuk panggilan future dan menangani hak yang dimiliki pelanggan dengan benar. Ini menguji bagian dari [Skenario: Layanan Anda memvalidasi pelanggan baru](#).
3. Setelah memverifikasi akun pengujian pada langkah sebelumnya, masukkan akun ke dalam aplikasi Anda. Misalnya, Anda dapat meminta pelanggan uji mengisi formulir untuk membuat pengguna baru. Atau, berikan mereka langkah selanjutnya untuk mendapatkan akses ke aplikasi SaaS Anda. Ini menguji bagian dari [Skenario: Layanan Anda memvalidasi pelanggan baru](#).
4. Jika tidak ada hak yang dikembalikan dari operasi GetEntitlements API, baik selama orientasi atau dalam pass verifikasi Anda yang sedang berlangsung, aplikasi Anda harus

mengelola akses dan pengalaman dengan benar untuk pengguna yang tidak berhak. Tes ini [Skenario: Layanan Anda menangani permintaan pelanggan](#).

5. Uji perubahan langganan. Verifikasi bahwa aplikasi Anda menangani skenario berhenti berlangganan, langganan yang berhasil, dan gagal berlangganan dengan benar. Tes ini [Skenario: Pantau perubahan langganan pengguna](#).
6. Setelah Anda menyelesaikan semua persyaratan integrasi dan menguji solusinya, beri tahu tim AWS Marketplace Operasi. Mereka kemudian akan menguji solusinya dengan memverifikasi bahwa Anda telah berhasil memanggil operasi `GetEntitlements` API dan cukup memasukkan pelanggan baru.

Setelah integrasi dan pengujian selesai, Anda dapat melakukan tinjauan akhir dan mencantumkan produk Anda di depan umum AWS Marketplace. Untuk informasi selengkapnya, lihat [Membuat produk SaaS di AWS Marketplace](#). Anda juga dapat membatalkan langganan pengujian Anda dengan mengisi Formulir Permintaan Pengembalian Dana. Untuk informasi selengkapnya tentang membatalkan langganan, lihat [the section called “Meminta pengembalian dana produk”](#)

Mengintegrasikan kontrak pay-as-you-go SaaS Anda dengan produk dengan AWS Marketplace

Mengintegrasikan produk Anda dengan AWS Marketplace adalah salah satu langkah. [Membuat produk SaaS di AWS Marketplace](#) Untuk mengintegrasikan produk kontrak perangkat lunak Anda sebagai layanan (SaaS) AWS Marketplace, Anda harus menulis kode dan menunjukkan bahwa ia dapat merespons dengan sukses beberapa skenario pelanggan. Bagian berikut menjelaskan skenario ini, menjelaskan cara menanggapi, dan memberikan gambaran umum tentang pengujian integrasi Anda.

Note

Sebelum Anda mulai, pastikan Anda telah memilih model harga yang tepat untuk perangkat lunak Anda sebagai produk layanan (SaaS). AWS Marketplace Untuk informasi selengkapnya, lihat [Merencanakan produk SaaS Anda](#).

Topik

- [Skenario: Layanan Anda memvalidasi pelanggan baru](#)
- [Skenario: Layanan Anda menangani permintaan pelanggan](#)

- [Skenario: Penggunaan meter](#)
- [Skenario: Memantau perubahan hak pengguna](#)
- [Menguji kontrak SaaS Anda dengan integrasi pay-as-you-go](#)

Skenario: Layanan Anda memvalidasi pelanggan baru

Ketika pelanggan berlangganan produk Anda, mereka diarahkan ke URL pendaftaran Anda, yang merupakan permintaan HTTP POST dengan token sementara `amzn-marketplace-token`. Tanggapi permintaan ini dengan cara berikut:

1. Tukarkan token dengan `CustomerIdentifierCustomerAWSAccountId`, dan `ProductCode` dengan memanggil operasi [ResolveCustomer](#) API di AWS Marketplace Metering Service.
2. Verifikasi langganan dan kuantitas (jika ada) yang dapat diakses pelanggan dengan memanggil [GetEntitlements](#) tindakan di AWS Marketplace Entitlement Service.
3. Pertahankan `CustomerIdentifierCustomerAWSAccountId`, dan `ProductCode` dalam sistem Anda untuk panggilan future. Simpan apakah pelanggan memiliki langganan yang valid, bersama dengan informasi apa pun yang Anda butuhkan tentang pelanggan.
4. Sebagai tanggapan atas permintaan tersebut, Anda harus menunjukkan pengalaman penggunaan pertama pengguna Anda (sebagaimana berlaku untuk layanan Anda).

Skenario: Layanan Anda menangani permintaan pelanggan

Ketika pelanggan membuat permintaan ke layanan Anda, Anda harus menanggapi skenario berikut dengan tindakan atau pesan yang sesuai:

- Mereka tidak memiliki ID pelanggan di sistem Anda. Ini berarti mereka belum berlangganan. Anda harus memberi mereka pesan yang menjelaskan cara berlangganan.
- Mereka memiliki ID pelanggan, dan operasi `GetEntitlements` API mengembalikan hak yang sesuai. Dalam skenario ini, Anda harus memenuhi permintaan.
- Mereka memang memiliki ID pelanggan, tetapi operasi `GetEntitlements` API tidak mengembalikan hak, atau jumlah yang tidak cukup untuk memenuhi permintaan. Dalam skenario ini, Anda harus menentukan cara menangani akses dan mengelola pengalaman mereka.

Skenario: Penggunaan meter

Ketika pelanggan mulai menggunakan layanan Anda, Anda harus mengirim catatan pengukuran setiap jam. Untuk detail tentang cara meteran, lihat [Mengkonfigurasi pengukuran untuk penggunaan dengan langganan SaaS](#).

Kami menyarankan Anda menggunakan AWS CloudTrail untuk memantau aktivitas untuk memastikan bahwa informasi penagihan sedang dikirim ke AWS. Ingatlah hal berikut saat mengirim catatan pengukuran:

- Permintaan pengukuran tidak digandakan pada jam tersebut.
- Catatan yang dikirim setiap jam bersifat kumulatif.
- Kami sangat menyarankan sebagai praktik terbaik bahwa, bahkan jika tidak ada catatan dalam satu jam terakhir, Anda mengirim catatan pengukuran setiap jam, dengan penggunaan 0.

Skenario: Memantau perubahan hak pengguna

Siapkan antrean Amazon Simple Queue Service (Amazon SQS), dan berlangganan topik Amazon SNS produk Anda—ada dua topik SNS, satu untuk perubahan hak dan satu untuk perubahan langganan. Informasi topik Anda disertakan dalam pesan email yang Anda terima dari tim Operasi AWS Marketplace Penjual saat Anda membuat produk. Untuk informasi selengkapnya, lihat [Membuat produk SaaS di AWS Marketplace](#). Dengan berlangganan topik SNS Anda, Anda menerima pemberitahuan tentang perubahan langganan pelanggan, termasuk menyediakan atau mencabut akses untuk pelanggan tertentu.

Note

Topik SNS Nama Sumber Daya Amazon (ARN) untuk perubahan langganan terlihat seperti. `arn:aws:sns:us-east-1:<account id>:aws-mp-subscription-notification-<product code>` Topik SNS ARN untuk perubahan hak terlihat seperti. `arn:aws:sns:us-east-1:<account id>:aws-mp-entitlement-notification-<product code>`

Pemberitahuan yang harus Anda tanggapi adalah sebagai berikut:

- `entitlement-updated`(dalam topik SNS hak) - Hak pelanggan telah berubah, dan Anda harus memanggil operasi `GetEntitlements` API untuk melihat status baru. Perbarui toko pelanggan

Anda, dan, jika berlaku (misalnya, kontrak pelanggan telah berakhir), ikuti praktik Anda untuk mematikan sumber daya pelanggan, mengikuti kebijakan retensi Anda.

- `subscribe-success`(dalam topik SNS berlangganan) - Pelanggan berlangganan, dan Anda dapat berhasil mengukur ID pelanggan mereka.
- `unsubscribe-pending`(dalam topik SNS berlangganan) - Pelanggan sedang dalam proses berhenti berlangganan. Anda harus mengirim catatan pengukuran terakhir.
- `unsubscribe-success`(dalam topik SNS berlangganan) - Pelanggan telah berhenti berlangganan. Catatan pengukuran untuk pelanggan tidak akan lagi diterima. Ikuti praktik Anda untuk mematikan sumber daya pelanggan, mengikuti kebijakan retensi Anda.
- `subscribe-fail`(dalam topik SNS berlangganan) - Langganan pelanggan gagal. Anda tidak boleh meteran terhadap ID pelanggan mereka atau mengaktifkan sumber daya atas nama pelanggan.

Note

Untuk informasi tambahan, lihat [Memeriksa hak menggunakan AWS Marketplace Entitlement Service](#).

Menguji kontrak SaaS Anda dengan integrasi pay-as-you-go

Setelah Anda mengintegrasikan kontrak Anda dengan pay-as-you-go produk AWS Marketplace, Anda harus melakukan pengujian mendalam untuk memastikan bahwa integrasi berhasil. Prosedur berikut menguraikan langkah-langkah untuk memverifikasi integrasi produk Anda.

Note

Gunakan akun Anda sendiri untuk berlangganan produk Anda dan uji apakah integrasi berhasil. Harga dapat dikurangi sementara sehingga Anda dapat menguji alur pembelian tanpa menimbulkan biaya tinggi di akun tersebut. Untuk informasi selengkapnya tentang mengurangi harga sementara atau mengizinkan akun pengujian tambahan mengakses produk Anda, hubungi tim [Operasi AWS Marketplace Penjual](#).

Setelah produk Anda diluncurkan, layanan harus terus menanggapi skenario ini untuk pelanggan baru.

1. Gunakan akun yang diizinkan untuk menguji pengalaman pelanggan dengan mendapatkan kontrak untuk produk Anda.
2. Setelah akun memiliki kontrak, pastikan bahwa akun dialihkan ke URL pendaftaran, dan bahwa pengalihan adalah permintaan POST yang menyertakan token sementara. Pastikan aplikasi Anda mempertahankan ID pelanggan untuk panggilan future dan menangani hak yang dimiliki pelanggan dengan benar. Ini menguji bagian dari [Skenario: Layanan Anda memvalidasi pelanggan baru](#).
3. Setelah memverifikasi akun pengujian pada langkah sebelumnya, masukkan akun ke dalam aplikasi Anda. Misalnya, Anda dapat meminta pelanggan uji mengisi formulir untuk membuat pengguna baru. Atau, berikan mereka langkah selanjutnya untuk mendapatkan akses ke aplikasi SaaS Anda. Ini menguji bagian dari [Skenario: Layanan Anda memvalidasi pelanggan baru](#).
4. Jika tidak ada hak yang dikembalikan dari operasi GetEntitlements API, baik selama orientasi atau dalam pass verifikasi Anda yang sedang berlangsung, aplikasi Anda harus mengelola akses dan pengalaman dengan benar untuk pengguna yang tidak berhak. Tes ini [Skenario: Layanan Anda menangani permintaan pelanggan](#).
5. Setelah pelanggan pengujian onboard, buat permintaan yang akan mengirim catatan pengukuran AWS untuk tujuan penagihan dengan menggunakan operasi BatchMeterUsage API di AWS Marketplace Metering Service Tes ini [Skenario: Penggunaan meter](#).
6. Uji perubahan langganan. Verifikasi bahwa aplikasi Anda menangani skenario berhenti berlangganan, langganan yang berhasil, dan gagal berlangganan dengan benar. Tes ini [Skenario: Memantau perubahan hak pengguna](#).
7. Setelah Anda menyelesaikan semua persyaratan integrasi dan menguji solusinya, beri tahu tim Operasi AWS Marketplace Penjual. Mereka kemudian akan menguji solusinya dengan memverifikasi bahwa Anda telah berhasil memanggil operasi GetEntitlements API dan cukup memasukkan pelanggan baru. Mereka juga akan memverifikasi bahwa Anda telah berhasil mengirim catatan terukur dengan operasi BatchMeterUsage API.

Setelah integrasi dan pengujian selesai, Anda dapat melakukan tinjauan akhir dan mencantumkan produk Anda di depan umum AWS Marketplace. Lihat informasi yang lebih lengkap di [Membuat produk SaaS di AWS Marketplace](#).

Menerapkan integrasi SaaS tanpa server

Setelah Anda membuat solusi SaaS, Anda mengintegrasikannya dengan daftar yang sesuai. AWS Marketplace Untuk melakukan itu, Anda dapat menggunakan AWS CloudFormation template untuk

membuat integrasi tanpa server. Integrasi ini memungkinkan Anda untuk mendaftarkan pelanggan, memberikan dan mencabut akses, memperbarui hak pelanggan, dan melaporkan penggunaan terukur.

Untuk informasi lebih lanjut tentang membuat integrasi SaaS tanpa server, lihat [Integrasikan SaaS Anda dengan referensi Integrasi SaaS Tanpa Server](#), lab yang menjelaskan cara menerapkan template dan menyelesaikan tugas terkait. AWS CloudFormation

Merencanakan produk SaaS Anda

Sebelum Anda menambahkan perangkat lunak Anda sebagai produk layanan (SaaS) AWS Marketplace, Anda harus terlebih dahulu melakukan beberapa perencanaan. Langkah ini sangat penting untuk keberhasilan produk Anda. Kurangnya perencanaan dapat mengakibatkan masalah penagihan atau Anda mungkin harus membuat ulang produk Anda. AWS Marketplace Bagian berikut menunjukkan kepada Anda bagaimana merencanakan produk SaaS,

Important

Sebagian besar pengaturan produk Anda tidak dapat diubah setelah Anda mengonfigurasinya. Jika Anda perlu mengubahnya setelah produk dibuat AWS Marketplace, Anda mungkin perlu membuat produk baru dengan pengaturan yang benar.

Topik

- [Rencanakan harga Anda](#)
- [Rencanakan integrasi penagihan Anda](#)
- [Rencanakan integrasi Amazon SNS Anda](#)
- [Rencanakan bagaimana pelanggan akan mengakses produk Anda](#)

Rencanakan harga Anda

Ada tiga model harga untuk produk SaaS. AWS Marketplace Memilih model harga yang tepat untuk produk Anda adalah keputusan paling penting yang akan Anda buat saat merencanakan produk Anda. Memilih model harga yang salah dapat membuat Anda mundur berminggu-minggu. Model penetapan harga menentukan opsi pembayaran untuk pelanggan Anda dan kode integrasi penagihan yang perlu Anda tulis, uji, dan terapkan. Untuk informasi tentang berbagai jenis model penetapan harga, lihat harga [produk SaaS](#).

Note

Semua model harga SaaS mendukung uji coba gratis. Untuk informasi lebih lanjut, lihat uji [coba gratis SaaS](#).

Rencanakan integrasi penagihan Anda

Salah satu manfaat memiliki produk SaaS adalah mengkonsolidasikan AWS Marketplace penagihan. Untuk memanfaatkan manfaat ini, Anda harus berintegrasi dengan AWS Marketplace Metering Service atau AWS Marketplace Entitlement Service, tergantung pada model harga yang Anda pilih. Kedua layanan ini membantu Anda memastikan bahwa pelaporan tagihan dan penggunaan Anda akurat.

Setelah Anda merencanakan integrasi Anda, Anda harus menguji integrasi dengan produk Anda sebelum ditayangkan. Untuk informasi selengkapnya tentang integrasi dan pengujian, lihat [Mengakses Layanan AWS Marketplace Pengukuran dan Hak APIs](#).

Rencanakan integrasi Amazon SNS Anda

Ada dua topik Amazon Simple Notification Service (Amazon SNS) yang dapat Anda berlangganan untuk produk SaaS Anda. Untuk informasi selengkapnya, lihat pemberitahuan [SaaS](#). Pesan-pesan ini dapat membantu Anda secara terprogram menangani perubahan langganan dan kontrak yang diprakarsai oleh AWS atau oleh pelanggan Anda. Notifikasi Amazon SNS dapat menjadi pemicu terprogram yang memungkinkan pelanggan untuk mendaftar akun baru di situs web pendaftaran produk Anda. Mereka juga dapat menolak pelanggan dengan langganan kedaluwarsa mengakses produk Anda. Anda memiliki opsi untuk bagaimana pelanggan Anda menerima pemberitahuan tergantung pada bagaimana Anda memprogram penanganan pemberitahuan ini.

Rencanakan bagaimana pelanggan akan mengakses produk Anda

Bagian ini menjelaskan cara membuat produk Anda dapat diakses oleh pembeli.

Rencanakan situs web pendaftaran produk SaaS Anda

Pelanggan yang membeli produk SaaS Anda membutuhkan akses ke sana. Anda harus merencanakan dan menerapkan bagaimana Anda ingin pelanggan Anda mengakses produk. Produk SaaS mendukung opsi akses berikut:

- Peluncuran Cepat
- AWS PrivateLink
- Situs web produk Anda sendiri

Untuk memvalidasi AWS Marketplace pelanggan menggunakan situs web pendaftaran Anda, lihat orientasi pelanggan [SaaS](#).

Menggunakan Quick Launch bagi pelanggan untuk mengakses produk Anda

Gunakan opsi penyebaran Quick Launch untuk mengurangi waktu dan sumber daya yang diperlukan pembeli untuk mengonfigurasi, menyebarkan, dan meluncurkan produk Anda. Quick Launch mengurangi jumlah situs yang harus dikunjungi pembeli selama proses berlangsung. Untuk informasi selengkapnya, lihat [Mengkonfigurasi Peluncuran Cepat](#).

Menggunakan pelanggan AWS PrivateLink untuk mengakses produk SaaS Anda

Anda dapat menggunakannya [Mengirimkan produk Anda melalui VPC Amazon menggunakan AWS PrivateLink](#) untuk mengonfigurasi layanan Anda sebagai layanan endpoint Amazon Virtual Private Cloud (Amazon VPC). Pelanggan Anda dapat membuat titik akhir VPC dan mengakses perangkat lunak Anda di seluruh jaringan virtual. AWS Cloud Atau, Anda dapat memberikan akses ke produk perangkat lunak Anda melalui situs web yang Anda miliki dan pelihara, dengan pelanggan membuat koneksi di internet.

Menggunakan situs web Anda sendiri

Produk SaaS Anda di-host di lingkungan Anda dan harus diakses melalui internet melalui titik akhir publik yang Anda kelola dan pelihara, seperti situs web. Biasanya, Anda memiliki situs web yang digunakan pelanggan untuk mendaftar produk Anda, masuk untuk menggunakan produk, dan mengakses dukungan untuk produk Anda.

Pedoman produk SaaS untuk AWS Marketplace

AWS Marketplace memelihara pedoman berikut untuk semua produk dan penawaran perangkat lunak sebagai layanan (SaaS) AWS Marketplace untuk mempromosikan platform yang aman, terjamin, dan dapat dipercaya bagi pelanggan kami. Bagian berikut memberikan pedoman untuk produk SaaS di AWS Marketplace

Semua produk dan metadata terkait ditinjau ketika dikirimkan untuk memastikan bahwa mereka memenuhi atau melampaui pedoman saat ini AWS Marketplace . Pedoman ini ditinjau dan

disesuaikan untuk memenuhi persyaratan keamanan kami yang terus berkembang. Selain itu, AWS Marketplace terus meninjau produk untuk memverifikasi bahwa mereka memenuhi setiap perubahan pada pedoman ini. Jika produk tidak sesuai, kami mungkin mengharuskan Anda memperbarui produk Anda dan dalam beberapa kasus produk Anda mungkin sementara tidak tersedia untuk pelanggan baru sampai masalah teratasi.

Topik

- [Pedoman pengaturan produk](#)
- [Persyaratan informasi pelanggan](#)
- [Pedoman penggunaan produk](#)
- [Pedoman arsitektur](#)

Pedoman pengaturan produk

Semua produk SaaS harus mematuhi pedoman pengaturan produk berikut:

- Dimensi harga tidak dapat dibatasi hanya untuk penawaran pribadi. Pembeli harus dapat berlangganan salah satu dimensi harga pada produk publik.
- Setidaknya satu dimensi harga harus memiliki harga lebih dari \$0,00.
- Semua dimensi harga harus berhubungan dengan perangkat lunak aktual dan tidak dapat menyertakan produk atau layanan lain yang tidak terkait dengan perangkat lunak.
- Produk SaaS yang ditawarkan secara eksklusif di AWS GovCloud (US) Wilayah harus termasuk GovCloud di suatu tempat dalam judul produk.

Persyaratan informasi pelanggan

Semua produk SaaS harus mematuhi persyaratan informasi pelanggan berikut:

- Produk SaaS harus ditagih seluruhnya melalui dimensi yang tercantum pada AWS Marketplace
- Anda tidak dapat mengumpulkan informasi pembayaran pelanggan untuk produk SaaS Anda kapan saja, termasuk informasi kartu kredit dan rekening bank.
- Halaman pendaftaran SaaS harus menyertakan bidang input untuk alamat email pembeli. Anda dapat menyertakan bidang tambahan seperti:
 - Nama
 - Kode Pos

- Nomor telepon
- Informasi perusahaan
- Preferensi pengaturan produk

Jika Anda berencana untuk menggunakan beberapa bahasa, Anda harus memberikan tampilan berbahasa Inggris dari halaman pendaftaran.

Pedoman penggunaan produk

Semua produk SaaS harus mematuhi pedoman penggunaan produk berikut:

- Setelah berlangganan produk di AWS Marketplace, pelanggan harus dapat membuat akun dalam aplikasi SaaS Anda dan mendapatkan akses ke konsol web. Jika pelanggan tidak dapat mengakses aplikasi dengan segera, Anda harus memberikan pesan dengan instruksi khusus tentang kapan mereka akan mendapatkan akses. Ketika akun telah dibuat, pelanggan harus dikirim pemberitahuan yang mengonfirmasi bahwa akun mereka telah dibuat bersama dengan langkah selanjutnya yang jelas.
- Jika pelanggan sudah memiliki akun di aplikasi SaaS, mereka harus memiliki kemampuan untuk masuk dari halaman arahan pemenuhan.
- Pelanggan harus dapat melihat status langganan mereka dalam aplikasi SaaS, termasuk kontrak atau informasi penggunaan langganan yang relevan.
- Pelanggan harus dapat dengan mudah mendapatkan bantuan dengan masalah seperti: menggunakan aplikasi, pemecahan masalah, dan meminta pengembalian uang (jika ada). Opsi kontak Dukungan harus ditentukan pada halaman arahan pemenuhan.
- Perangkat lunak dan metadata produk tidak boleh berisi bahasa yang mengarahkan pengguna ke platform cloud lain, produk tambahan, layanan upsell, atau penawaran uji coba gratis yang tidak tersedia. AWS Marketplace

Untuk informasi tentang uji coba gratis untuk produk SaaS, lihat. [Membuat penawaran uji coba gratis SaaS di AWS Marketplace](#)

- Jika produk Anda merupakan add-on untuk produk lain atau produk ISV lain, deskripsi produk Anda harus menunjukkan bahwa itu memperluas fungsionalitas produk lain dan bahwa tanpanya, produk Anda memiliki utilitas yang sangat terbatas. Misalnya, Produk ini memperluas fungsionalitas dan tanpa itu, produk ini memiliki utilitas yang sangat terbatas<product name>. Harap dicatat bahwa mungkin memerlukan lisensi sendiri untuk fungsionalitas penuh dengan daftar ini. <product name>

Pedoman arsitektur

Topik berikut mencantumkan dan menjelaskan pedoman arsitektur untuk produk SaaS.

Topik

- [Pedoman](#)
- [Membuat diagram arsitektur](#)

Pedoman

Note

Pedoman berikut berlaku efektif per 1 Mei 2025.

- Anda dapat mempublikasikan semua arsitektur SaaS.
- Produk yang digunakan pada AWS menerima penunjukan khusus dalam hasil AWS Marketplace pencarian dan halaman detail produk mereka. AWS Marketplace Untuk mempertimbangkan produk Anda sebagai “dikerahkan AWS,” produk Anda harus berjalan AWS sepenuhnya. Ini termasuk aplikasi dan pesawat kontrol.

Pesawat aplikasi dapat berjalan di AWS akun penjual, AWS akun pembeli, atau keduanya. Untuk informasi lebih lanjut, lihat whitepaper [Control plane vs. Application plane](#).

Layanan pihak ketiga yang digunakan oleh produk untuk mengirimkan, menyimpan, atau memproses data aplikasi — kecuali jaringan pengiriman konten (CDNs), sistem nama domain (DNSs), dan penyedia identitas perusahaan (IdPs) — juga harus dijalankan sepenuhnya. AWS

Note

Data aplikasi adalah data yang dimiliki atau dihasilkan untuk pembeli.

Agan atau gateway yang digunakan oleh produk untuk keamanan, pemantauan, replikasi data, atau migrasi dapat berjalan di lingkungan milik pembeli di luar AWS, termasuk di tempat, tetapi harus mengirim data hanya untuk AWS penyimpanan dan analisis.

Anda harus menyertakan diagram arsitektur untuk ditinjau. Anda tidak dapat membuat diagram publik. Untuk informasi lebih lanjut, lihat [Membuat diagram arsitektur](#) di bagian selanjutnya.

- Penjual dapat mempublikasikan produk yang tidak sepenuhnya berjalan AWS.
- Aplikasi yang membutuhkan sumber daya dalam infrastruktur pembeli harus mengikuti pedoman ini:
 - Untuk dianggap sebagai produk SaaS dan bukan layanan terkelola, pesawat kontrol Anda—sebagaimana didefinisikan dalam whitepaper [Dasar-dasar AWS Arsitektur SaaS—harus berada di infrastruktur](#) yang Anda kelola. Untuk informasi selengkapnya, lihat whitepaper [SaaS vs. Managed Service Provider](#).
 - Dalam deskripsi produk, Anda harus memberi tahu pelanggan bahwa jika mereka dikenakan biaya AWS infrastruktur terpisah dari AWS Marketplace transaksi mereka, mereka harus membayar biaya tersebut.
 - Anda harus menyediakan sumber daya dengan cara yang aman, seperti menggunakan AWS Security Token Service (AWS STS) atau AWS Identity and Access Management (IAM).
 - Anda harus mengikuti [prinsip hak istimewa paling sedikit](#) saat membuat instruksi penggunaan atau templat penerapan yang memberikan izin ke aplikasi Anda.
 - Anda harus memberikan dokumentasi tambahan yang menjelaskan semua AWS layanan yang disediakan, pernyataan kebijakan IAM, dan bagaimana peran IAM atau pengguna digunakan dan digunakan di akun pelanggan.
 - Anda harus memberikan instruksi atau templat penyebaran yang memungkinkan pembeli untuk menyebarkan sumber daya yang diperlukan di akun mereka AWS .
 - Jika Anda menyediakan AWS CloudFormation template (CFTs) untuk menyebarkan sumber daya ke AWS akun pembeli, mereka harus mematuhi [AWS Marketplace kebijakan untuk CFTs](#). Anda harus mempublikasikannya CFTs sebagai bagian dari daftar SaaS Anda dengan mengikuti metode yang disediakan saat Anda mengaktifkan opsi [penyebaran SaaS Quick Launch](#) untuk pembeli Anda. SaaS Quick Launch memudahkan pembeli Anda untuk mengonfigurasi solusi SaaS Anda.
 - Jika Amazon Machine Images (AMIs) disebarkan ke AWS akun pembeli, mereka harus mematuhi [AWS Marketplace kebijakan untuk AMIs](#). Anda AMIs harus [melewati pemindai AMI](#) di Portal AWS Marketplace Manajemen (portal penjual). Saat [meminta produk Anda untuk dipublikasikan](#), Anda juga harus [menghubungi AWS Marketplace operasi](#) dan memberikan bukti hasil pemindaian.
 - Jika gambar kontainer disebarkan ke AWS akun pembeli, gambar harus mematuhi [AWS Marketplace kebijakan untuk kontainer](#). Gambar kontainer Anda dapat di-host di luar AWS,

tetapi harus [dipindai di Amazon Elastic Container Registry \(Amazon ECR\) Registry ECR](#) [dan bebas](#) dari kerentanan kritis. Saat [meminta produk Anda untuk dipublikasikan](#), Anda juga harus [menghubungi AWS Marketplace operasi](#) dan memberikan bukti bahwa wadah melewati pemindaian.

- Berhasil menelepon AWS Marketplace APIs dari Akun AWS yang terdaftar sebagai penyedia dan mengajukan permintaan penerbitan SaaS. Model penetapan harga SaaS menentukan mana yang APIs harus disebut:
 - Kontrak SaaS — [GetEntitlements](#) di AWS Marketplace Entitlement Service
 - Kontrak SaaS dengan konsumsi — [GetEntitlements](#) di AWS Marketplace Entitlement Service dan [BatchMeterUsage](#) di AWS Marketplace Metering Service.
 - Langgan SaaS — [BatchMeterUsage](#) di AWS Marketplace Metering Service.
- Produk SaaS yang ditawarkan secara eksklusif di AWS GovCloud (US) Daerah harus menjelaskan batas-batas arsitektur antara lain Wilayah AWS dan AWS GovCloud (US) Wilayah, kasus penggunaan untuk produk, dan beban kerja yang tidak direkomendasikan untuk produk tersebut.

[Untuk informasi lebih lanjut tentang arsitektur SaaS, lihat whitepaper SaaS Architecture Fundamentals.](#) AWS

Membuat diagram arsitektur

Untuk menerima penunjukan khusus tempat produk Anda digunakan AWS, [perbarui detail arsitektur produk Anda di Portal](#) AWS Marketplace Manajemen. Pilih pola hosting yang digunakan AWS dan unggah diagram arsitektur yang AWS mengulas. Untuk pola hosting yang AWS Marketplace mempertimbangkan untuk digunakan AWS, lihat [Pedoman](#) di bagian sebelumnya. Jika pola hosting Anda berubah, Anda harus memperbarui detail arsitektur produk Anda.

Note

- Anda tidak dapat membuat diagram Anda publik.
- Anda dapat mempublikasikan produk yang tidak berjalan sepenuhnya AWS.

Gunakan kriteria berikut saat membuat diagram:

- Kelompokkan dan beri label komponen sebagai bagian dari [bidang aplikasi atau bidang kontrol](#).

- Untuk komponen apa pun di luar AWS yang merupakan bagian dari logika bisnis inti produk Anda, kelompokkan dengan bidang aplikasi.
- Komponen dapat mewakili detail tingkat rendah (misalnya, instance komputasi dan subnet jaringan), atau layanan tingkat tinggi (misalnya, platform analisis data).
- Komponen tidak perlu mengidentifikasi nama AWS layanan atau AWS non-layanan yang digunakan.
- Tempatkan komponen di mana mereka berjalan secara logis. Misalnya, di AWS akun penjual, AWS akun pembeli, AWS lingkungan non-penjual, atau lingkungan lain.
- Untuk replikasi data atau produk migrasi beban kerja, sertakan semua lingkungan sumber dan target yang didukung.

Note

Diagram arsitektur yang Anda gunakan untuk memperbarui detail arsitektur produk SaaS Anda tidak dipublikasikan dan tidak tersedia untuk umum untuk pembeli.

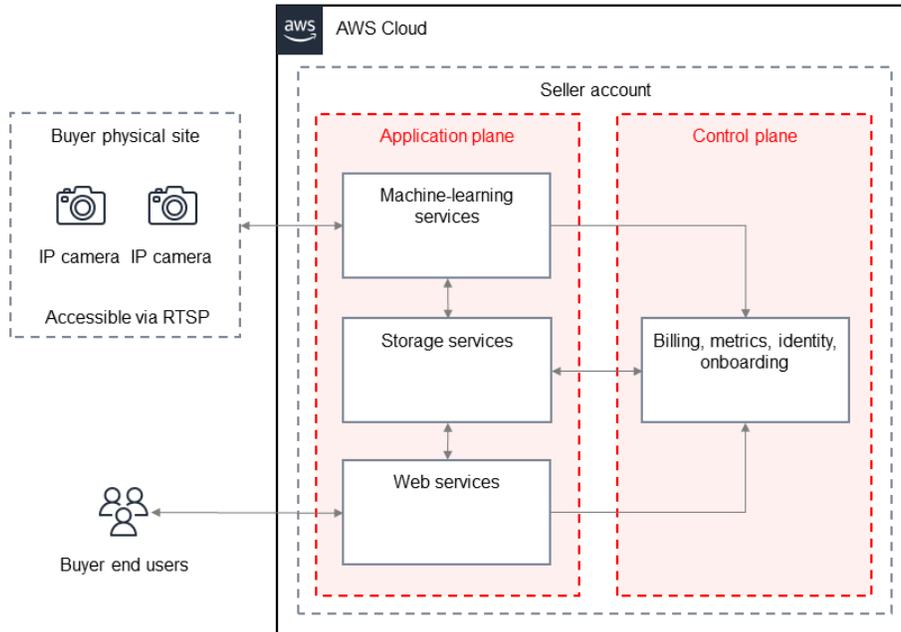
Tingkat detail

Anda dapat membuat diagram tingkat tinggi yang menunjukkan komponen sistem utama, termasuk aliran data dasar, dan berfokus pada bidang aplikasi dan layanan bidang kontrol. Atau, Anda dapat membuat diagram terperinci tingkat rendah yang memecah setiap komponen, menunjukkan koneksi tertentu, dan menyertakan spesifikasi teknis dengan tingkat detail yang berbeda.

Diagram berikut menunjukkan arsitektur aplikasi SaaS analisis video hipotetis. Masing-masing menunjukkan tingkat detail yang berbeda. Keduanya bisa diterima. Gunakan mereka sebagai contoh untuk tingkat detail untuk dimasukkan dalam diagram Anda sendiri.

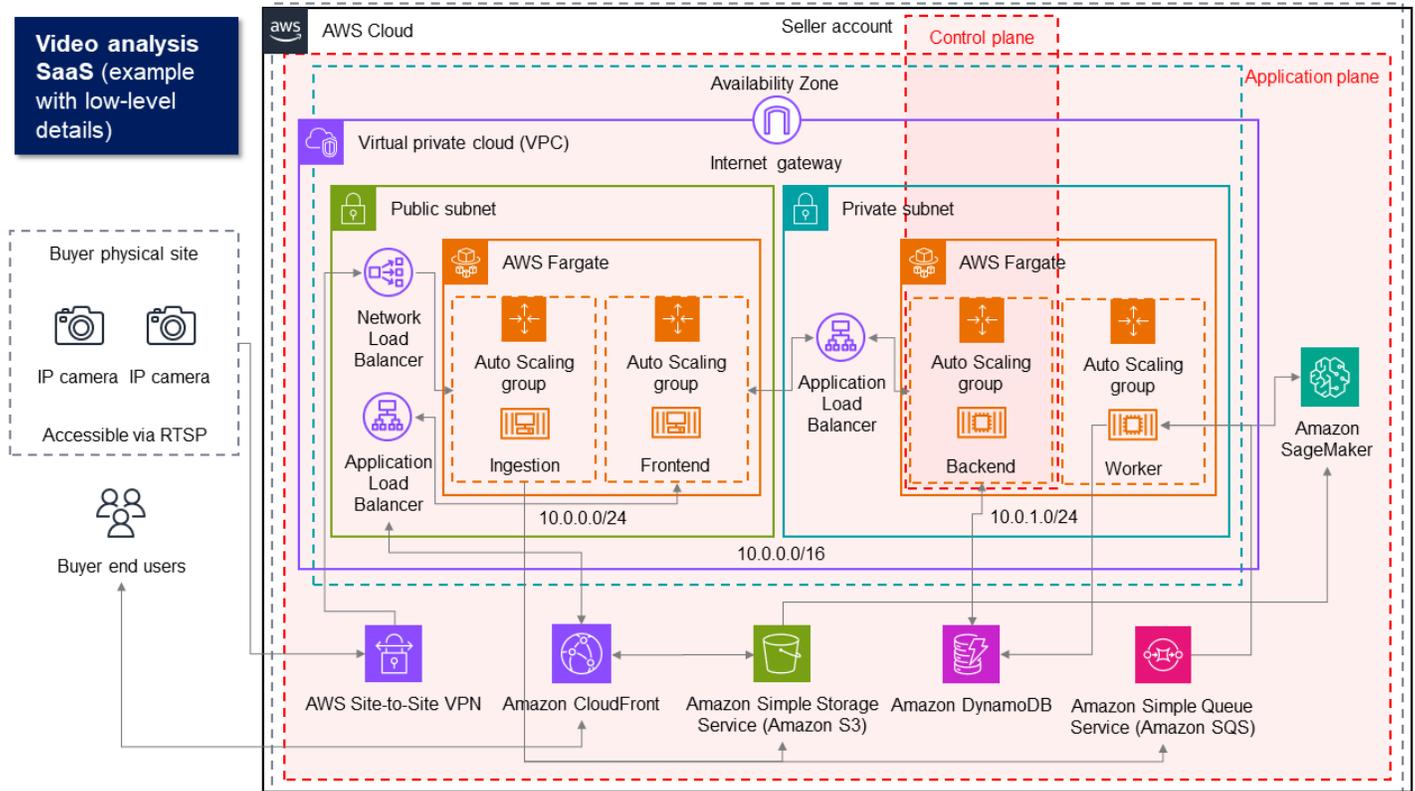
Berikut ini adalah contoh diagram tingkat tinggi.

Video analysis SaaS (example with high-level services)



Berikut ini adalah contoh diagram rinci tingkat rendah.

Video analysis SaaS (example with low-level details)



Untuk informasi lebih lanjut, lihat [Apa itu Diagram Arsitektur?](#) . Setelah membuat diagram, perbarui detail arsitektur Anda di Portal AWS Marketplace Manajemen (AMMP). Lihat informasi yang lebih lengkap di [Perbarui detail arsitektur](#).

Harga produk SaaS di AWS Marketplace

Setelah pembeli membeli produk perangkat lunak Anda sebagai layanan (SaaS) AWS Marketplace, AWS Marketplace memberi Anda pengenalan penagihan mereka. Anda menggunakan pengenalan penagihan untuk memanggil AWS Marketplace Entitlement Service dan AWS Marketplace Metering Service. Kemudian, pelanggan mengakses produk di AWS lingkungan Anda atau melalui koneksi titik akhir virtual private cloud (VPC) yang Anda buat. Topik ini menyediakan daftar model harga SaaS untuk AWS Marketplace.

Note

Semua model harga SaaS mendukung uji coba gratis. Untuk informasi lebih lanjut, lihat [uji coba gratis SaaS](#).

Model harga SaaS

Model penentuan harga	Deskripsi
Langganan SaaS	pay-as-you-go Model tempat kami menagih pembeli untuk penggunaan produk SaaS Anda setiap jam. Untuk informasi selengkapnya, lihat Harga untuk langganan SaaS .
Kontrak SaaS	Pembeli ditagih terlebih dahulu untuk penggunaan perangkat lunak Anda, atau Anda dapat menawarkan jadwal pembayaran yang fleksibel kepada mereka. Pelanggan juga dapat membayar untuk penggunaan tambahan di atas kontrak mereka. Untuk informasi selengkapnya, lihat Harga untuk kontrak SaaS .
Kontrak SaaS dengan pay-as-you-go	Pembeli ditagih terlebih dahulu untuk penggunaan perangkat lunak Anda, atau Anda dapat menawarkan jadwal pembayaran yang

Model penentuan harga	Deskripsi
	fleksibel kepada mereka. Pembeli juga ditagih tarif terukur tambahan untuk penggunaan di atas harga kontrak. Untuk informasi selengkapnya, lihat Harga untuk kontrak SaaS .

Note

Setelah Anda membuat iklan dan mempublikasikannya ke terbatas, Anda tidak dapat mengubah model harga.

[Untuk membuat produk SaaS Anda tersedia AWS Marketplace, putuskan apakah Anda ingin menawarkan model harga langganan SaaS atau model penetapan harga kontrak SaaS.](#)

Harga untuk langganan SaaS

Untuk langganan perangkat lunak sebagai layanan (SaaS), AWS Marketplace tagihan pelanggan Anda berdasarkan catatan pengukuran yang Anda kirimkan kepada kami. Semua biaya harus diukur dan dilaporkan setiap jam dari perangkat lunak yang digunakan di akun pelanggan. Semua penggunaan kemudian dihitung setiap bulan dan ditagih setiap bulan menggunakan mekanisme yang sama dengan AWS Marketplace penawaran berbasis AMI. Kemampuan kami untuk menagih pelanggan untuk penggunaan produk Anda bergantung pada penerimaan catatan pengukuran dari Anda. Anda bertanggung jawab untuk memastikan bahwa catatan pengukuran produk Anda berhasil dikirim dan diterima.

Sebelum Anda dapat mempublikasikan produk SaaS dengan harga berlangganan, Anda harus melakukan hal berikut:

1. Buat produk SaaS baru di Portal Manajemen AWS Marketplace, pilih Langganan SaaS Baru.
2. Lengkapi bidang di tab Umum dengan informasi yang diperlukan. Catat kode produk.
3. Pada tab Harga, di bawah Tetapkan Harga, pilih Kategori yang menjelaskan harga produk Anda paling akurat. Kategori harga muncul untuk pelanggan di situs AWS Marketplace web. Anda dapat memilih dari Bandwidth (GBps, MBps), Data (GB, MB, TB), Host (jam), Permintaan, atau Pengguna (jam). Jika tidak ada kategori standar yang sesuai dengan kebutuhan Anda, Anda dapat memilih kategori Unit yang lebih umum.

Selanjutnya, tentukan Dimensi Harga Anda. Setiap Dimensi Harga mewakili fitur atau layanan yang dapat Anda tetapkan harga per unit. Contoh dimensi termasuk pengguna, host yang dipindai, dan GB log yang dicerna. Anda dapat menentukan hingga 24 dimensi. Untuk setiap dimensi yang Anda tentukan, Anda harus menambahkan informasi berikut:

- **Dimension API Name** - Nama API yang digunakan saat mengirim catatan pengukuran ke file. [AWS Marketplace Metering Service](#) Nama ini menunjukkan dimensi mana yang digunakan pelanggan Anda. Nama ini terlihat dalam laporan penagihan. Nama tidak harus ramah pembaca karena Anda satu-satunya yang memiliki akses ke laporan Anda. Setelah membuat bucket, Anda tidak dapat mengubah namanya.
- **Deskripsi Dimensi** — Pernyataan yang menghadap pelanggan yang menggambarkan dimensi produk. Deskripsi dapat tidak lebih dari 70 karakter dan harus user-friendly. Contoh deskripsi adalah Administrator per jam, dan bandwidth Per Mbps disediakan. Setelah produk diterbitkan, Anda dapat mengubah deskripsi ini.
- **Tingkat Dimensi** — Biaya perangkat lunak per unit FCP untuk produk ini, dalam USD. Bidang ini mendukung delapan tempat desimal.

Saat langganan SaaS berakhir

Pelanggan dapat berhenti berlangganan dari produk langganan SaaS Anda melalui AWS Management Console Poin-poin penting dari proses akhir langganan SaaS meliputi yang berikut:

1. Produk SaaS Anda dikirim `unsubscribe-pending` pemberitahuan melalui topik Amazon SNS untuk pelanggan tersebut.
2. Anda memiliki satu jam untuk mengukur sisa penggunaan untuk pelanggan.
3. Setelah jam ini, Anda menerima `unsubscribe-success` pemberitahuan. Pada titik ini, Anda tidak dapat lagi mengirim catatan pengukuran untuk pelanggan ini.

Terserah Anda untuk memutuskan bagaimana Anda ingin menonaktifkan fungsionalitas dalam produk SaaS Anda untuk pelanggan yang berhenti berlangganan. Misalnya, produk Anda mungkin menyelesaikan pekerjaan pelanggan yang ada tetapi mencegah mereka membuat pekerjaan. Anda mungkin ingin menampilkan pesan kepada pelanggan bahwa penggunaannya telah dinonaktifkan. Pelanggan dapat berlangganan kembali produk Anda melalui AWS Marketplace.

Saat langganan SaaS dibatalkan

Poin penting dari proses pembatalan langganan SaaS meliputi:

1. Pelanggan dapat membatalkan langganan mereka ke produk langganan SaaS Anda di halaman Perangkat Lunak Marketplace Anda di AWS Marketplace situs web.

Produk SaaS Anda dikirim pemberitahuan melalui topik Amazon SNS untuk pelanggan tersebut.

2. Anda memiliki satu jam untuk mengukur sisa penggunaan untuk pelanggan.
3. Anda memberi tahu pelanggan dari produk Anda bahwa pembatalan sedang berlangsung. Jika pelanggan menunjukkan bahwa mereka ingin membatalkan melalui produk Anda, arahkan pelanggan ke AWS Marketplace. Untuk menjamin bahwa tidak akan ada biaya future, pelanggan harus memastikan pembatalan tersebut. AWS Marketplace

Harga untuk kontrak SaaS

Untuk kontrak perangkat lunak sebagai layanan (SaaS), pelanggan memulai pembelian perangkat lunak Anda dan menandatangani perjanjian dengan Anda. Berdasarkan perjanjian, pelanggan berhak atas jumlah tertentu penggunaan produk SaaS Anda. AWS Marketplace mengkomunikasikan hak-hak ini ke aplikasi SaaS Anda. Hal ini dilakukan melalui AWS Marketplace Entitlement Service. Saat menggunakan model penetapan harga Kontrak SaaS, aplikasi Anda tidak pernah mengirimkan catatan pengukuran. Sebaliknya, itu memverifikasi hak dengan memanggil. AWS Marketplace Entitlement Service Anda menentukan kategori penggunaan, dimensi, dan panjang kontrak.

AWS Marketplace menagih pelanggan Anda di muka atau dengan jadwal pembayaran yang Anda tentukan, berdasarkan kontrak antara Anda dan pelanggan Anda. Setelah itu, mereka berhak menggunakan sumber daya tersebut. Untuk penggunaan tambahan di atas kontrak mereka, perangkat lunak Anda perlu melaporkan penggunaan dan AWS Marketplace menagih pelanggan Anda berdasarkan catatan pengukuran yang diterima oleh kami melalui. AWS Marketplace Metering Service

Sebelum Anda dapat mempublikasikan produk SaaS dengan harga kontrak, Anda harus melakukan hal berikut:

1. Buat produk SaaS baru di Portal Manajemen AWS Marketplace, dan pilih Kontrak SaaS Baru.
2. Lengkapi bidang di tab Umum dengan informasi yang diperlukan. Catat kode produk.
3. Pada tab Harga:
 - a. Untuk Set Pricing, pilih Durasi Kontrak yang ingin Anda tawarkan kepada pelanggan. Anda dapat memasukkan harga yang berbeda untuk setiap durasi kontrak. Anda dapat memilih satu atau lebih opsi berikut: Bulanan, 1 tahun, 2 Tahun, dan 3 Tahun. Jika Anda membuat

- penawaran pribadi, Anda dapat memilih durasi khusus dalam beberapa bulan (hingga 60 bulan).
- b. Untuk Pilih jenis kontrak yang ingin Anda tawarkan, pilih bagaimana Anda ingin pelanggan dapat membeli produk Anda dari opsi berikut:
 - Pembeli dapat memilih satu atau lebih opsi yang ditawarkan — Pelanggan dapat memilih kuantitas untuk setiap dimensi harga yang Anda tawarkan.
 - Pembeli dapat memilih satu tingkat dari beberapa tingkatan yang ditawarkan — Pelanggan memilih tingkat dari opsi yang mencakup serangkaian fitur, layanan, dan jumlah penggunaan yang berbeda.
 - c. Pilih kategori unit penggunaan yang menjelaskan harga produk Anda paling akurat. Kategori harga muncul untuk pelanggan di situs AWS Marketplace web. Anda dapat memilih dari Bandwidth (GBps, MBps), Data (GB, MB, TB), Host (jam), Permintaan, atau Pengguna (jam). Jika tidak ada kategori standar yang sesuai dengan kebutuhan Anda, Anda dapat memilih kategori Unit yang lebih umum.
4. Setelah Anda memilih kategori, tentukan Dimensi Harga Anda. Setiap Dimensi Harga mewakili fitur atau layanan yang dapat Anda tetapkan per satuan harga. Contoh dimensi adalah pengguna, host yang dipindai, dan GB log yang dicerna. Untuk setiap dimensi yang Anda tentukan, Anda menambahkan nama, deskripsi, harga, dan API nama. Nama, harga, dan deskripsi ditampilkan kepada pelanggan. Anda menggunakan API nama untuk melacak dan melaporkan dengan AWS Marketplace sebagai berikut:
- Memanggil [AWS Marketplace Entitlement Service](#) untuk mengambil dimensi yang telah dibeli pelanggan Anda.
 - Memanggil [AWS Marketplace Metering Service](#) untuk menunjukkan dimensi mana yang digunakan pelanggan.

Untuk setiap dimensi harga dalam kontrak Anda, Anda dapat memilih untuk membiarkan pelanggan membayar saat mereka pergi untuk penggunaan tambahan dimensi itu di atas kontrak mereka. Anda juga dapat menambahkan dimensi tambahan tanpa harga kontrak yang hanya dikonsumsi pelanggan dengan membayar saat mereka pergi.

Saat menggunakan wizard untuk membuat kontrak untuk produk SaaS Anda, Anda harus menentukan bidang berikut untuk dimensi harga Anda:

- APINama Dimensi — Nama yang digunakan saat memanggil HakAPI. Nama ini terlihat dalam laporan penagihan dan laporan yang tidak menghadap ke luar. Panjang maksimum untuk API nama adalah 15 karakter. Setelah Anda menetapkan nama, itu tidak dapat diubah.

- Nama Tampilan Dimensi: — Nama dimensi yang menghadap pelanggan. Nama ini akan membantu pelanggan memahami dimensi produk. Nama harus user-friendly, dan panjang maksimumnya adalah 24 karakter. Nilai ini dapat diubah.
- Deskripsi Dimensi: — Deskripsi dimensi yang menghadap pelanggan yang memberikan informasi tambahan tentang dimensi produk. Panjang maksimum untuk deskripsi adalah 70 karakter.
- Dimensi - Harga Bulanan - Biaya perangkat lunak per unit untuk opsi 1 bulan untuk dimensi ini. Bidang ini mendukung tiga angka desimal.
- Dimensi - Harga 1 Tahun - Biaya perangkat lunak per unit untuk opsi 12 bulan untuk dimensi ini. Bidang ini mendukung tiga angka desimal. Ini bukan biaya bulanan. Harga harus mencerminkan harga pengisian satu kali 12 bulan.
- Dimensi - Harga 2 Tahun - Biaya perangkat lunak per unit untuk opsi 24 bulan untuk dimensi ini. Bidang ini mendukung tiga angka desimal.
- Dimensi - Harga 3 Tahun - Biaya perangkat lunak per unit untuk opsi 36 bulan untuk dimensi ini. Bidang ini mendukung tiga angka desimal.

Contoh: Aplikasi penyimpanan data

	Harga bulanan	Harga 12 bulan	Harga 24 bulan	ay-as-you-go Harga P untuk penggunaan tambahan
Data tidak terenkripsi (GB)	\$1,50/GB	\$16,00/GB	\$30.00/GB	\$0,1/GB per jam
Data terenkripsi (GB)	\$1,55/GB	\$16,60/GB	\$31.20/GB	\$0,1/GB per jam

Contoh: Produk pemantauan log

	Harga bulanan	Harga 12 bulan	ay-as-you-go Harga P untuk penggunaan tambahan
Dasar (10 host dipantau, 5 kontainer dipantau)	\$100	\$1000	
Standar (20 host dipantau, 10 kontainer dipantau)	\$200	\$2000	
Pro (40 host dipantau, 20 kontainer dipantau)	\$400	\$4000	
Host tambahan dipantau per jam			\$0,1
Kontainer tambahan dipantau per jam			\$0,2

 Note

Harga bisa untuk jangka waktu berikut: 1 bulan, 12 bulan, 24 bulan, atau 36 bulan. Anda dapat memilih untuk menawarkan satu atau lebih opsi ini untuk produk Anda. Durasi harus sama di setiap dimensi. Misalnya, asumsikan bahwa Anda memiliki `ReadOnlyUsers` dan `AdminUsers` dimensi. Jika Anda menawarkan harga tahunan untuk `ReadOnlyUsers`, Anda harus menawarkan harga `AdminUsers` tahunan juga.

Peningkatan kontrak SaaS

Pelanggan dapat meningkatkan kontrak ke salah satu nilai yang lebih tinggi kecuali untuk jangka waktu yang lebih lama. Misalnya, mereka dapat meningkatkan ke jumlah yang lebih tinggi atau hak bernilai lebih tinggi. Pelanggan diberikan kredit prorata untuk kontrak mereka yang ada. Pelanggan

tidak dapat mengurangi ukuran kontrak yang ada. Mereka hanya dapat mengurangi ukuran saat pembaruan, atau membatalkan pembaruan mereka.

Hak diverifikasi oleh produk SaaS Anda, yang membuat panggilan ke AWS Marketplace Entitlement Service

Perpanjangan otomatis

Ketika pelanggan membeli produk Anda melalui AWS Marketplace penggunaan kontrak SaaS, mereka dapat menyetujui perpanjangan otomatis ketentuan kontrak. Pelanggan terus membayar hak setiap bulan atau selama 1, 2, atau 3 tahun. Pelanggan selalu memiliki opsi untuk mengubah pengaturan pembaruan. Mereka dapat membatalkan pembaruan atau memperbarui kontrak untuk jumlah dan durasi yang berbeda.

Ketika kontrak SaaS berakhir

Produk kontrak SaaS memiliki kontrak kedaluwarsa. Ketika kontrak berakhir, peristiwa berikut terjadi:

1. Produk SaaS Anda menerima entitlement-updated pemberitahuan yang menunjukkan hak pembeli telah berubah. AWS Marketplace Entitlement Service Mengembalikan respon kosong.
2. Anda memiliki 1 jam untuk mengukur sisa penggunaan untuk pelanggan. Setelah waktu ini berlalu, Anda tidak dapat lagi mengirim catatan pengukuran untuk pelanggan ini.

Ketika kontrak SaaS dibatalkan

Poin-poin penting dari proses pembatalan kontrak SaaS meliputi:

1. Pelanggan dapat meminta pembatalan dan pengembalian dana untuk produk kontrak SaaS. AWS Dukungan

Pelanggan harus meminta pengembalian dana dalam waktu 48 jam. AWS Dukungan

Pengembalian dana penuh atau prorata biasanya diberikan dalam 3-5 hari kerja.

2. Produk SaaS Anda dikirim pemberitahuan melalui topik SNS Amazon untuk pelanggan tersebut.
3. Anda memiliki waktu satu jam untuk mengirim catatan pengukuran akhir untuk pelanggan untuk biaya penggunaan tambahan.
4. Anda memberi tahu pelanggan dari produk Anda bahwa pembatalan sedang berlangsung. Jika pelanggan menunjukkan bahwa mereka ingin membatalkan melalui produk Anda, arahkan

pelanggan ke AWS Marketplace. Untuk menjamin bahwa tidak akan ada biaya future, pelanggan harus memastikan pembatalan tersebut. AWS Marketplace

Membuat penawaran uji coba gratis SaaS di AWS Marketplace

Sebagai AWS Marketplace penjual, Anda dapat membuat penawaran uji coba gratis perangkat lunak sebagai layanan (SaaS) di Portal Manajemen AWS Marketplace (AMMP). Pelanggan dapat mengevaluasi produk perangkat lunak sebelum membuat keputusan pembelian besar dengan menggunakan opsi uji coba gratis SaaS. Setelah pelanggan berlangganan produk Anda, produk Anda melakukan pemeriksaan hak dengan cara yang sama seperti halnya untuk pelanggan berbayar.

Masing-masing hanya Akun AWS dapat menggunakan uji coba gratis untuk produk SaaS satu kali. Jumlah penggunaan gratis yang diberikan selama uji coba gratis tidak dibagikan di seluruh akun tertaut dalam AWS organisasi. Akun tertaut yang berbeda dalam satu akun pembayar utama dapat membuat uji coba gratis masing-masing.

Note

- Jika Anda menggunakan Layanan Pengiriman Data Penjual (SDDS), Anda akan menerima [laporan uji coba detail Perjanjian](#) di bucket Amazon Simple Storage Service. Laporan tersebut mencakup rincian perjanjian seperti nama pelanggan dan ID, ID penawaran, dan tanggal mulai dan berakhir perjanjian.
- Penjual juga menerima [notifikasi Amazon Simple Notification Service \(Amazon SNS\)](#) saat langganan baru dibuat. Notifikasi Amazon SNS menyertakan `isFreeTrialTermPresent` bendera untuk mengidentifikasi perjanjian uji coba gratis.
- Selain itu, pelanggan yang berlangganan uji coba gratis Anda diarahkan ke URL pendaftaran Anda dengan token tambahan, `x-amzn-marketplace-offer-type=free-trial`. Anda dapat menggunakan token untuk menciptakan pengalaman pendaftaran unik bagi pelanggan yang menggunakan uji coba gratis Anda.

Membuat penawaran uji coba gratis SaaS

Penjual dapat membuat penawaran uji coba gratis SaaS di Portal Manajemen AWS Marketplace (AMMP).

Untuk membuat penawaran uji coba gratis SaaS

1. Masuk ke [Portal Manajemen AWS Marketplace](#).
2. Pada Portal Manajemen AWS Marketplace, pilih salah satu:
 - Buat atau kelola penawaran
 - Tab Penawaran
3. Di halaman Penawaran, pilih tab Uji coba gratis Publik untuk meninjau semua uji coba gratis SaaS.
4. Pilih Buat penawaran uji coba gratis. Penjual dapat membuat satu penawaran uji coba gratis SaaS per setiap produk SaaS publik.
5. Untuk dasar-dasar Penawaran, pilih Produk Anda dan kemudian pilih Berikutnya.
6. Dalam pengaturan uji coba gratis:
 - a. Masukkan jumlah hari untuk durasi uji coba gratis (hari) Anda.

Durasi uji coba gratis berkisar antara 7-90 hari.
 - b. Lihat dimensi Produk dari penawaran umum Anda yang ada.

Anda tidak dapat mengubah dimensi produk untuk uji coba gratis langganan SaaS.

Anda dapat mengatur batas kuantitas per setiap dimensi untuk uji coba gratis kontrak SaaS, dan Hapus atau Tambahkan dimensi.
7. Lihat perjanjian Layanan.

Untuk versi EULA, Anda dapat memilih Kontrak Standar untuk AWS Marketplace atau EULA Kustom, lalu pilih Penawaran ulasan.
8. Verifikasi dan tinjau semua informasi untuk penawaran, lalu pilih Buat penawaran.

Membatalkan penawaran uji coba gratis SaaS

Penjual dapat membatalkan penawaran uji coba gratis kapan saja dari Portal Manajemen AWS Marketplace.

Untuk membatalkan penawaran uji coba gratis SaaS

1. Masuk ke [Portal Manajemen AWS Marketplace](#).

2. Pada Portal Manajemen AWS Marketplace, pilih salah satu:
 - Buat atau kelola penawaran
 - Tab Penawaran
3. Pada halaman Penawaran, pilih penawaran.
4. Pilih Lihat penawaran.
5. Pilih Batalkan penawaran.

Setelah penawaran dibatalkan, perjanjian aktif untuk penawaran ini aktif hingga kedaluwarsa. Perjanjian baru untuk penawaran yang dibatalkan tidak dapat dibuat.

Pelanggan orientasi ke produk SaaS Anda melalui AWS Marketplace

Dengan langganan perangkat lunak sebagai layanan (SaaS) dan kontrak SaaS, pelanggan Anda berlangganan produk Anda melalui AWS Marketplace tetapi mengakses produk di lingkungan Anda. AWS Setelah berlangganan produk, pelanggan Anda diarahkan ke situs web yang Anda buat dan kelola sebagai bagian dari produk SaaS Anda untuk mendaftarkan akun mereka dan mengonfigurasi produk.

Saat membuat daftar produk SaaS Anda, Anda memberikan URL ke halaman arahan pendaftaran Anda. Kami menggunakan URL tersebut untuk mengarahkan pelanggan ke halaman arahan pendaftaran Anda setelah mereka berlangganan. Pada halaman arahan pendaftaran perangkat lunak Anda, Anda mengumpulkan informasi yang diperlukan untuk membuat akun bagi pelanggan. Kami merekomendasikan untuk mengumpulkan alamat email pelanggan Anda jika Anda berencana untuk menghubungi mereka melalui email untuk pemberitahuan penggunaan.

Halaman landing pendaftaran harus dapat mengidentifikasi dan menerima `x-amzn-marketplace-token` token dalam data formulir dari AWS Marketplace pengenalan pelanggan untuk penagihan. Kemudian harus meneruskan nilai token tersebut ke AWS Marketplace Metering Service untuk menyelesaikan Akun AWS ID pelanggan unik, pengenalan pelanggan (tidak digunakan lagi setelah 31 Desember 2025), dan kode produk yang sesuai. Untuk contoh kode, lihat [ResolveCustomercontoh kode](#).

Note

Token pendaftaran diselesaikan ke pelanggan berlangganan tertentu dan setiap token yang dihasilkan memiliki jendela kedaluwarsa 4 jam. Selama pemanggil memanggil API dengan token yang sama, ia akan terus mengembalikan nilai respons yang sama sampai token kedaluwarsa.

Mengkonfigurasi produk SaaS Anda untuk menerima pembeli baru

Anda bertanggung jawab untuk mengonfigurasi perangkat lunak SaaS Anda dengan benar untuk menerima pelanggan baru dan mengukurnya dengan tepat. Proses berikut menguraikan satu cara yang disarankan untuk mengidentifikasi, menerapkan, dan mengukur akses pelanggan baru ke perangkat lunak Anda:

1. Ketika pelanggan mengunjungi halaman produk Anda di AWS Marketplace situs web, mereka memilih untuk berlangganan produk Anda.
2. Pelanggan Akun AWS berlangganan produk Anda. Ini berarti catatan langganan dan pengukuran yang dikirim dari produk Anda menjadi bagian dari AWS tagihan pelanggan.
3. Token pendaftaran dibuat untuk pelanggan yang berisi Akun AWS ID, pengenalan pelanggan, dan kode produk Anda.
4. Pelanggan diarahkan ke halaman arahan pendaftaran perangkat lunak Anda. Halaman ini harus dapat menerima token dengan pengenalan pelanggan.
5. Browser pelanggan mengirimkan POST permintaan ke URL halaman arahan pendaftaran perangkat lunak Anda. Permintaan berisi satu POST parameter `x-amzn-marketplace-token`, yang berisi token pendaftaran pelanggan. Dari perspektif situs web pendaftaran Anda, pelanggan telah mengirimkan formulir dengan parameter ini. Token pendaftaran adalah string buram. Jika jenis penawaran adalah uji coba gratis, parameter kedua, `x-amzn-marketplace-offer-type` dengan nilai `free-trial`, akan ditambahkan ke permintaan.
6. Untuk menukarkan token pendaftaran ini dengan Akun AWS ID pelanggan, pengenalan pelanggan, dan kode produk, situs web Anda harus menghubungi AWS [ResolveCustomer](#) Marketplace Metering Service. Untuk contoh `ResolveCustomer` panggilan, lihat [ResolveCustomer](#) contoh kode. Identifikasi pelanggan bukanlah Akun AWS ID pelanggan, tetapi bersifat universal di antara produk dan harus disimpan ke sumber internal sebagai bagian dari catatan pelanggan Anda. Kode produk adalah string unik untuk produk SaaS Anda yang AWS menyediakan untuk

Anda. Setiap AWS produk memiliki satu kode produk unik, yang diberikan kepada Anda saat pendaftaran.

7. Pelanggan diinstruksikan untuk membuat akun di produk Anda atau masuk ke akun yang ada.

 Note

Jika menyiapkan atau menautkan ke akun pelanggan yang ada di produk Anda memerlukan proses manual oleh tim Anda, Anda dapat menggunakan formulir kontak-kami untuk mengumpulkan informasi kontak pelanggan. Setelah mengumpulkan informasi kontak mereka dan menyelesaikan Akun AWS ID dan pengenalan pelanggan unik mereka (seperti yang diperoleh pada langkah 6), tampilkan pesan pemberitahuan untuk pelanggan. Dalam pemberitahuan, nyatakan bahwa akun mereka sedang disiapkan dan minta mereka menunggu Anda untuk menghubungi mereka. Berikan pelanggan dengan waktu penyelesaian yang diharapkan dan informasi kontak Anda. Kirim juga pesan email ke pelanggan dengan detail yang sama.

8. Pelanggan sekarang masuk ke situs web Anda menggunakan kredensial khusus untuk produk SaaS tersebut. Dalam database akun Anda, Anda dapat memiliki entri untuk setiap pelanggan. Database akun Anda harus memiliki kolom untuk Akun AWS ID. Verifikasi bahwa tidak ada akun lain di sistem Anda yang membagikan Akun AWS ID.
9. Selama proses pendaftaran penjual, Anda berlangganan topik Amazon SNS yang memberi tahu Anda saat pelanggan berlangganan atau berhenti berlangganan produk Anda. Ini adalah pemberitahuan Amazon SNS dalam format JSON yang memberi tahu Anda tentang tindakan pelanggan:
 - Pemberitahuan hak — Untuk produk dengan model harga yang menyertakan kontrak, Anda diberi tahu saat pembeli membuat kontrak baru, meningkatkannya, memperbaruinya, atau kedaluwarsa. Database akun Anda harus memiliki kolom tambahan untuk status berlangganan. Untuk informasi selengkapnya, lihat [Topik Amazon SNS: aws-mp-entitlement-notification](#).
 - Pemberitahuan berlangganan — Untuk produk dengan model harga apa pun, termasuk kontrak dan langganan, Anda diberi tahu saat pembeli berlangganan atau berhenti berlangganan suatu produk. Untuk informasi selengkapnya, lihat [Topik Amazon SNS: aws-mp-subscription-notification](#).

Sebaiknya gunakan Amazon Simple Queue Service (Amazon Simple Queue Service) untuk menangkap pesan-pesan ini. Setelah Anda menerima pemberitahuan berlangganan dengan `subscribe-success`, akun pelanggan siap untuk pengukuran. Rekaman yang Anda kirim sebelum pemberitahuan ini tidak diukur. Untuk informasi tentang cara melakukannya, lihat [Langkah 2: Berikan izin ke topik Amazon SNS untuk mengirim pesan ke antrean Amazon SQS di Panduan Pengembang Layanan Pemberitahuan Sederhana Amazon](#).

 Note

Jangan aktifkan langganan produk kecuali Anda menerima `subscribe-success` pemberitahuan.

10. Gunakan Akun AWS ID yang disimpan dalam database Anda untuk mengukur penggunaan melalui AWS Marketplace Metering Service atau periksa hak melalui AWS Marketplace Entitlement Service

Keamanan dan pemesanan

Sebagai penjual, Anda bertanggung jawab untuk mempercayai hanya Akun AWS IDs yang segera dikembalikan dari AWS atau yang telah ditandatangani oleh sistem Anda. Kami menyarankan Anda menyelesaikan token pendaftaran segera karena mungkin kedaluwarsa setelah sekitar satu jam. Setelah Anda menyelesaikan token pendaftaran, simpan Akun AWS ID sebagai atribut yang ditandatangani pada sesi browser pelanggan hingga pendaftaran selesai.

Notifikasi Amazon SNS untuk produk SaaS

Untuk menerima pemberitahuan, Anda berlangganan topik Amazon Simple Notification Service (Amazon SNS) AWS Marketplace untuk diberikan kepada Anda selama pembuatan produk. Topik memberikan pemberitahuan tentang perubahan langganan pelanggan dan hak kontrak untuk produk Anda. Ini memungkinkan Anda mengetahui kapan harus menyediakan dan mencabut akses untuk pelanggan tertentu.

Note

Selama proses pembuatan produk, Anda akan menerima Nama Sumber Daya Amazon (ARN) yang sebenarnya ke topik SNS. Sebagai contoh: `arn:aws:sns:us-east-1:123456789012:aws-mp-subscription-notification-PRODUCTCODE`

Topik Amazon SNS berikut tersedia untuk produk perangkat lunak sebagai layanan (SaaS):

- [Topik Amazon SNS: `aws-mp-entitlement-notification`](#)— Topik ini memberi tahu Anda ketika pembeli membuat kontrak baru, meningkatkannya, memperbaruinya, atau kedaluwarsa. Ini hanya tersedia untuk produk dengan model harga yang mencakup kontrak (juga dikenal sebagai Kontrak SaaS dan Kontrak SaaS dengan Konsumsi (Kelebihan)).
- [Topik Amazon SNS: `aws-mp-subscription-notification`](#)— Topik ini memberi tahu Anda ketika pembeli berlangganan atau berhenti berlangganan produk dan menyertakan penawaran `offer-identifier` untuk pribadi dan bendera uji coba gratis untuk uji coba gratis SaaS. Ini tersedia untuk semua model harga, termasuk kontrak dan langganan (juga dikenal sebagai Langganan SaaS, Kontrak SaaS, dan Kontrak SaaS dengan Konsumsi (Kelebihan)).

Untuk mempelajari lebih lanjut tentang skenario di mana Anda menanggapi pemberitahuan ini, lihat topik berikut:

- [Mengintegrasikan produk langganan SaaS Anda dengan AWS Marketplace](#)
- [Mengintegrasikan produk kontrak SaaS Anda dengan AWS Marketplace](#)
- [Mengintegrasikan kontrak pay-as-you-go SaaS Anda dengan produk dengan AWS Marketplace](#)

Topik Amazon SNS: `aws-mp-entitlement-notification`

Setiap pesan dalam `aws-mp-entitlement-notification` topik memiliki format berikut.

```
{
  "action": "<action-name>",
  "customer-identifier": " X01EXAMPLE",
  "product-code": "n0123EXAMPLEXXXXXXXXXXXX",
}
```

`<action-name>` Akan selalu begitu `entitlement-updated`.

Note

- Untuk pesan hak, terlepas dari tindakannya (baru, peningkatan, pembaruan, atau kedaluwarsa), pesannya sama. Panggilan berikutnya GetEntitlement diperlukan untuk menemukan konten pembaruan.
- [Untuk Kontrak SaaS dengan Konsumsi \(Kelebihan\), penjual diberikan topik SNS. `aws-mp-subscription-notification`](#) Ini adalah pemberitahuan tambahan yang diterima penjual ketika mereka menambahkan harga berlebih. Ketika penjual memperoleh pelanggan baru, alih-alih hanya mendapatkan entitlement-updated (yang dapat merujuk pada segala jenis tindakan), penjual menerima pesan berlangganan yang menunjukkan bahwa ini adalah pelanggan baru.
- Untuk future dated agreement (FDAs), topik ini dimulai pada tanggal mulai perjanjian (dan bukan tanggal tanda perjanjian). Ini juga dimulai ketika perubahan berikutnya terjadi dalam hak, seperti pembatalan, penggantian, pembaruan, atau berakhirnya perjanjian.

Produk dengan harga kontrak (termasuk kontrak dengan pay-as-you-go) harus menanggapi pesan-pesan ini. Untuk informasi selengkapnya tentang cara merespons, lihat [Skenario: Pantau perubahan langganan pengguna](#).

Topik Amazon SNS: `aws-mp-subscription-notification`

Setiap pesan dalam `aws-mp-subscription-notification` topik memiliki format berikut.

```
{
  "action": "<action-name>",
  "customer-identifier": " X01EXAMPLE",
  "product-code": "n0123EXAMPLEXXXXXXXXXXXX",
  "offer-identifier": "offer-abcexample123",
  "isFreeTrialTermPresent": "true"
}
```

`offer-identifier` Ini termasuk dalam pemberitahuan hanya ketika tindakan `subscribe-success` atau `subscribe-fail`. Itu tidak termasuk dalam notifikasi saat tindakan `unsubscribe-pending` atau `unsubscribe-success`. Untuk penawaran yang dibuat sebelum Januari 2024, pengenal ini disertakan dalam pemberitahuan hanya untuk penawaran pribadi. Untuk penawaran yang dibuat pada Januari 2024 dan yang lebih baru, pengenal ini disertakan dalam pemberitahuan untuk semua penawaran, termasuk penawaran pribadi dan penawaran publik.

Untuk informasi tentang jenis penawaran, lihat tanggapan dari [DescribeEntity API](#) atau visibilitas penawaran perjanjian di dasbor [perpanjangan Perjanjian](#).

 Note

Untuk [DescribeEntity API](#), jika Anda menemukan aspek penargetan Akun AWS dalam akun dari aturan penargetan untuk penawaran itu, itu adalah penawaran pribadi. Jika tidak ada aspek penargetan Akun AWS dalam akun dari aturan penargetan untuk penawaran itu, itu adalah penawaran umum.

`isFreeTrialTermPresentProperti` menunjukkan apakah langganan pembeli adalah uji coba gratis. Nilai JSON dari properti ini bukan tipe data boolean. Sebaliknya, nilai dikonversi ke tipe data string. Untuk informasi lebih lanjut, lihat uji [coba gratis SaaS](#).

`<action-name>` Akan bervariasi tergantung pada notifikasi. Tindakan yang mungkin adalah:

- `subscribe-success`— `subscribe-success` Pesan memberi sinyal ketika penjual dapat mulai mengirim catatan pengukuran. Jika [penawaran berbasis perjanjian](#) diterima oleh pembeli, pesan ini dikirim lagi dengan yang baru. `offer-identifier`
- `subscribe-fail`— Jika `subscribe-fail` pesan dihasilkan, pembayaran mungkin gagal meskipun pembeli telah beralih dari halaman arahan SaaS penjual AWS Marketplace ke penjual. Penjual harus menunggu `subscribe-success` pesan sebelum mengizinkan konsumsi produk.
- `unsubscribe-pending`— Ketika pembeli berhenti berlangganan, `unsubscribe-pending` pesan dikirim terlebih dahulu. Ini menunjukkan bahwa penjual memiliki waktu terbatas (sekitar satu jam) untuk mendapatkan catatan pengukuran akhir yang dikirim sebelum pembeli dibatalkan sepenuhnya.
- `unsubscribe-success`— `unsubscribe-success` Pesan menandakan selesainya pembatalan, setelah itu tidak ada catatan pengukuran lebih lanjut yang akan diterima.

 Note

- Jika pembeli berhenti berlangganan dan kemudian segera berhasil berlangganan ulang sebelum `unsubscribe-success` pesan terakhir dikirim, pesan terakhir tidak akan dikirim dan `unsubscribe-success` pesan akan dikirim sebagai `subscribe-success` gantinya.

- Untuk future dated agreement (FDAs), `subscribe-success` tindakan dimulai pada tanggal mulai perjanjian (dan bukan tanggal tanda perjanjian).

Produk dengan harga berlangganan (termasuk kontrak dengan `pay-as-you-go`) harus menanggapi pesan-pesan ini. Untuk informasi selengkapnya tentang cara merespons, lihat topik berikut:

- [Mengintegrasikan produk langganan SaaS Anda dengan AWS Marketplace](#)
- [Mengintegrasikan kontrak `pay-as-you-go` SaaS Anda dengan produk dengan AWS Marketplace](#)

Berlangganan antrian SQS ke topik SNS

Sebaiknya berlangganan antrian Amazon SQS ke topik SNS yang disediakan. Untuk petunjuk mendetail tentang cara membuat antrian SQS dan berlangganan antrian ke topik, lihat [Berlangganan antrian Amazon SQS ke topik Amazon SNS di Panduan Pengembang Layanan Pemberitahuan Sederhana Amazon](#).

Note

Anda hanya dapat berlangganan topik AWS Marketplace SNS dari yang Akun AWS digunakan untuk menjual produk. Namun, Anda dapat meneruskan pesan ke akun lain. Untuk informasi selengkapnya, lihat [Mengirim pesan Amazon SNS ke antrian Amazon SQS di akun lain di Panduan Pengembang](#) Layanan Pemberitahuan Sederhana Amazon.

Polling antrian SQS untuk pemberitahuan

Setelah Anda berlangganan antrian SQS Anda ke topik SNS, pesan disimpan dalam SQS. Anda harus menentukan layanan yang terus-menerus melakukan polling antrian, mencari pesan, dan menanganinya sesuai dengan itu.

Mengakses Layanan AWS Marketplace Pengukuran dan Hak APIs

Bagian ini menguraikan proses integrasi dengan AWS Marketplace Metering Service atau AWS Marketplace Entitlement Service, yang digunakan untuk memastikan bahwa penagihan dan pelaporan Anda untuk penggunaan pelanggan atas produk perangkat lunak sebagai layanan (SaaS) Anda akurat. Diasumsikan bahwa Anda telah mengirimkan produk langganan SaaS atau produk

kontrak SaaS yang telah diterbitkan ke negara bagian terbatas. Dalam keadaan terbatas, Anda dapat menggunakan akun pengujian untuk memverifikasi konfigurasi dan fungsi yang tepat tetapi produk Anda tidak tersedia untuk umum.

Note

Jika produk SaaS Anda terintegrasi dengan layanan AWS terkelola lain yang menangani pengukuran dengan cara yang berbeda (seperti Amazon SageMaker Ground Truth, atau AWS WAF), maka Anda tidak perlu berintegrasi dengan layanan pengukuran AWS Marketplace. Pengukuran untuk produk Anda seharusnya hanya terjadi dalam satu sistem untuk menghindari penagihan ganda pelanggan Anda.

Topik

- [Mengkonfigurasi pengukuran untuk penggunaan dengan langganan SaaS](#)
- [Memeriksa hak menggunakan AWS Marketplace Entitlement Service](#)
- [Daftar periksa integrasi produk SaaS](#)

Untuk informasi tentang pengaturan AWS CLI, bersama dengan kredensialnya, lihat [Mengonfigurasi AWS CLI dalam Panduan Pengguna.AWS Command Line Interface](#) [Jika Anda baru mengenal AWS Python SDK](#), lihat [Boto 3 Quickstart](#).

Mengkonfigurasi pengukuran untuk penggunaan dengan langganan SaaS

Untuk langganan perangkat lunak sebagai layanan (SaaS), Anda meteran untuk semua penggunaan, dan kemudian pelanggan ditagih AWS berdasarkan catatan pengukuran yang Anda berikan. Untuk kontrak SaaS, Anda hanya meteran untuk penggunaan di luar hak kontrak pelanggan. Bagian berikut memberikan informasi tentang cara mengonfigurasi pengukuran untuk penggunaan dengan produk SaaS.

Ketika aplikasi Anda mengukur penggunaan untuk pelanggan, aplikasi Anda menyediakan AWS dengan jumlah penggunaan yang masih harus dibayar. Pengukur aplikasi Anda untuk dimensi harga yang Anda tentukan saat membuat produk, seperti gigabyte yang ditransfer atau host dipindai dalam jam tertentu. Misalnya, jika Anda mengenakan biaya berdasarkan jumlah data yang dikirim ke aplikasi Anda, Anda dapat mengukur jumlah data dan mengirim catatan pengukuran yang sesuai satu jam sekali. AWS menghitung tagihan pelanggan menggunakan data pengukuran bersama dengan harga yang Anda berikan saat Anda membuat produk Anda.

Note

Secara opsional, Anda dapat membagi penggunaan di seluruh properti yang Anda lacak. Properti ini diekspos ke pembeli sebagai tag. Tag ini memungkinkan pembeli untuk melihat biaya mereka dibagi menjadi penggunaan oleh nilai tag. Misalnya, jika Anda mengenakan biaya oleh pengguna, dan pengguna memiliki Department properti, Anda dapat membuat alokasi penggunaan dengan tag yang memiliki kunci Department, dan satu alokasi per nilai. Ini tidak mengubah harga, dimensi, atau total penggunaan yang Anda laporkan, tetapi memungkinkan pelanggan Anda untuk melihat biaya mereka berdasarkan kategori yang sesuai dengan produk Anda. Untuk informasi selengkapnya, lihat [Penandaan terukur vendor \(Opsional\)](#).

Topik

- [Meter setiap jam](#)
- [Konfigurasi produk Anda untuk penggunaan meter](#)
- [Penandaan terukur vendor \(Opsional\)](#)

Meter setiap jam

Kami menyarankan Anda melaporkan penggunaan setiap jam untuk semua pelanggan Anda dalam batch hingga 25 sekaligus. AWS ini memberi pelanggan visibilitas granular sebanyak mungkin ke dalam penggunaan dan biaya mereka. Jika Anda menggabungkan penggunaan dalam periode waktu lebih dari satu jam (misalnya, satu hari), perhatikan pertimbangan berikut.

- AWS hanya dapat menagih pelanggan untuk penggunaan produk Anda setelah menerima catatan pengukuran dari Anda. Anda bertanggung jawab untuk memastikan bahwa catatan pengukuran produk Anda berhasil dikirim dan diterima. Anda dapat menggunakan AWS CloudTrail untuk memverifikasi catatan atau catatan yang Anda kirim akurat. Anda juga dapat menggunakan informasi untuk melakukan audit dari waktu ke waktu. Untuk informasi selengkapnya, lihat [Panggilan Logging AWS Marketplace Metering API dengan AWS CloudTrail](#).
- Jika ini adalah SaaS dengan model harga “Berlangganan” (bukan model penetapan harga “Kontrak” atau “Kontrak dengan Konsumsi”), maka pembeli dapat berhenti berlangganan kapan saja. Ketika pembeli memulai tindakan berhenti berlangganan ini, penjual akan menerima unsubscribe-pending pemberitahuan dan memiliki waktu 1 jam untuk mengirimkan semua penggunaan yang tidak dilaporkan sebelum pemberitahuan akhir. unsubscribe-success Apa

pun setelah pemberitahuan kedua tidak akan dihormati. Dua model harga lainnya memiliki durasi yang ditetapkan berdasarkan waktu berlangganan dan pembeli tidak dapat berhenti berlangganan selama itu. Mereka hanya dapat mematikan autorenewal. Pemberitahuan yang sama dikirim pada akhir durasi itu jika tidak autorenewing.

- Jika Anda tidak mengirim catatan pengukuran setiap jam dan ada aplikasi atau pemadaman jaringan, catatan Anda akan tertinggal lebih jauh. Hal ini dapat mengakibatkan penggunaan yang tidak dilaporkan jika aplikasi atau pemadaman jaringan dipulihkan setelah langganan berakhir.
- Bahkan jika tidak ada gunanya melaporkan, Anda dapat terus mengirim catatan pengukuran setiap jam dan mencatat jumlah 0 jika tidak ada gunanya melaporkan untuk jam itu. Perhatikan bahwa setelah Anda melaporkan penggunaan dimensi oleh pembeli, 0 atau lebih, Anda tidak dapat mengubah catatan tersebut. Oleh karena itu adalah praktik terbaik untuk melaporkan penggunaan untuk jam sebelumnya. Penggunaan yang dilaporkan terlihat di konsol [AWS Billing and Cost Management](#) pembeli selama siklus penagihan. Konsol menunjukkan setiap dimensi produk Anda dalam item baris terpisah dengan jumlah total unit yang dikonsumsi dan total biaya untuk item baris yang dikonsumsi untuk siklus penagihan itu atau sejauh ini, bahkan jika harga dimensi adalah \$0 per unit. Jika diaktifkan, [Laporan Biaya dan Penggunaan AWS \(CUR\)](#) menunjukkan detail ini termasuk tarif per unit. Laporan ini tidak diperbarui secara real time.
- Selama penerbitan, tim AWS Marketplace Operasi akan menguji bahwa aplikasi SaaS berhasil mengirimkan catatan pengukuran sebelum mengizinkan produk dipublikasikan. Biasanya, tim akan melakukan pendaftaran tiruan SaaS dan mengonfirmasi bahwa catatan pengukuran diterima.

Note

Jika produk SaaS Anda terintegrasi dengan layanan AWS terkelola lain yang menangani pengukuran dengan cara yang berbeda (seperti Amazon SageMaker Ground Truth, atau AWS WAF), maka Anda tidak perlu berintegrasi dengan layanan pengukuran AWS Marketplace. Pengukuran untuk produk Anda seharusnya hanya terjadi dalam satu sistem untuk menghindari penagihan ganda pelanggan Anda. Perhatikan bahwa AWS Marketplace tidak menerbitkan AWS WAF produk baru saat ini.

Konfigurasi produk Anda untuk penggunaan meter

Anda menggunakan `BatchMeterUsage` operasi di AWS Marketplace Metering Service untuk mengirimkan catatan pengukuran ke AWS. Ingatlah hal-hal berikut ini:

- Kami mengharuskan penjual untuk menggunakan batching dengan menggunakan BatchMeterUsage operasi.
- Kami menghapus duplikasi permintaan pengukuran pada jam tersebut.
 - Permintaan dideduplikasi per. product/customer/hour/dimension
 - Anda selalu dapat mencoba lagi permintaan apa pun, tetapi jika Anda mengukur untuk jumlah yang berbeda, jumlah asli ditagih.
 - Jika Anda mengirim beberapa permintaan untuk hal yang sama customer/dimension/hour, catatan tidak digabungkan.
- Penjual dapat mengirim catatan pengukuran dengan stempel waktu hingga 6 jam sebelumnya jika pelanggan berlangganan produk Anda. Jika pelanggan berhenti berlangganan, penjual harus mengirim catatan pengukuran dalam waktu 1 jam setelah pelanggan berhenti berlangganan.
- BatchMeterUsage muatan tidak boleh melebihi 1MB. Pilih jumlah catatan penggunaan yang akan dikirim dalam BatchMeterUsage permintaan sehingga Anda tidak melebihi ukuran muatan.
- Layanan Pengukuran AWS Marketplace tersedia di [AWS Marketplace titik akhir dan kuota yang Wilayah AWS tercantum dalam Referensi](#) Umum.AWS Secara default, Wilayah AS Timur (Virginia N.) diaktifkan untuk produk pengukuran SaaS saat Anda meminta produk Anda. Jika Anda berniat menggunakan Wilayah lain, hubungi tim [Operasi AWS Marketplace Penjual](#). Untuk informasi selengkapnya, lihat [BatchMeterUsage](#).

Untuk contoh kode, lihat [Contoh kode untuk integrasi produk SaaS](#).

Contoh: Pemindaian host

Produk Anda menganalisis perangkat keras komputasi untuk kerentanan keamanan yang diketahui. Pelanggan secara manual memulai atau menjadwalkan pemindaian ini dari instans Amazon Elastic Compute Cloud (Amazon EC2) mereka. Saat produk Anda melakukan pemindaian ini, itu menghitung jumlah host unik yang dipindai setiap jam. Dalam contoh ini, produk Anda menggunakan kategori Host. Anda dapat mendeklarasikan beberapa dimensi untuk jenis host yang dipindai. Misalnya, Anda dapat mengenakan harga yang berbeda untuk host kecil, menengah, dan besar.

Contoh: Analisis log

Produk SaaS Anda mencerna log yang dihasilkan oleh produk pelanggan, tren pelaporan, dan anomali. Saat pelanggan mengunggah log ke produk Anda, Anda mengukur jumlah data yang diterima dalam megabyte, gigabyte, atau terabyte. Pada menit kesepuluh setiap jam, pekerjaan cron membaca penggunaan ini untuk setiap pelanggan selama satu jam sebelumnya. Pekerjaan membangun laporan batch dan menggunakan BatchMeterUsage operasi untuk mengirimkannya

ke AWS. Dalam contoh ini, produk Anda menggunakan kategori Data. Produk Anda juga dapat mengukur jumlah data log yang disimpan selama satu jam tertentu. Dalam hal ini, produk Anda dapat mengukur sepanjang dua dimensi: data yang diterima dalam satu jam dan total data yang disimpan dalam satu jam. Anda dapat terus mengukur data yang disimpan hingga pelanggan menghapus data ini atau kedaluwarsa.

Penandaan terukur vendor (Opsional)

Penandaan yang diukur vendor membantu Vendor Perangkat Lunak Independen (ISVs) memberi pembeli wawasan yang lebih terperinci tentang penggunaan perangkat lunak mereka dan dapat membantu mereka melakukan alokasi biaya.

Ada banyak cara untuk menandai penggunaan perangkat lunak pembeli. Salah satu caranya adalah dengan terlebih dahulu bertanya kepada pembeli Anda apa yang ingin mereka lihat dalam alokasi biaya mereka. Kemudian Anda dapat membagi penggunaan di seluruh properti yang Anda lacak untuk akun pembeli. Contoh properti termasuk `Account ID`, `Business Unit`, `Cost Centers`, dan metadata relevan lainnya untuk produk Anda. Properti ini diekspos ke pembeli sebagai tag. Dengan menggunakan tag, pembeli dapat melihat biayanya dibagi menjadi penggunaan berdasarkan nilai tag di Konsol AWS Penagihan (<https://console.aws.amazon.com/costmanagement/>). Penandaan yang diukur vendor tidak mengubah harga, dimensi, atau total penggunaan yang Anda laporkan. Hal ini memungkinkan pelanggan Anda untuk melihat biaya mereka berdasarkan kategori yang sesuai dengan produk Anda.

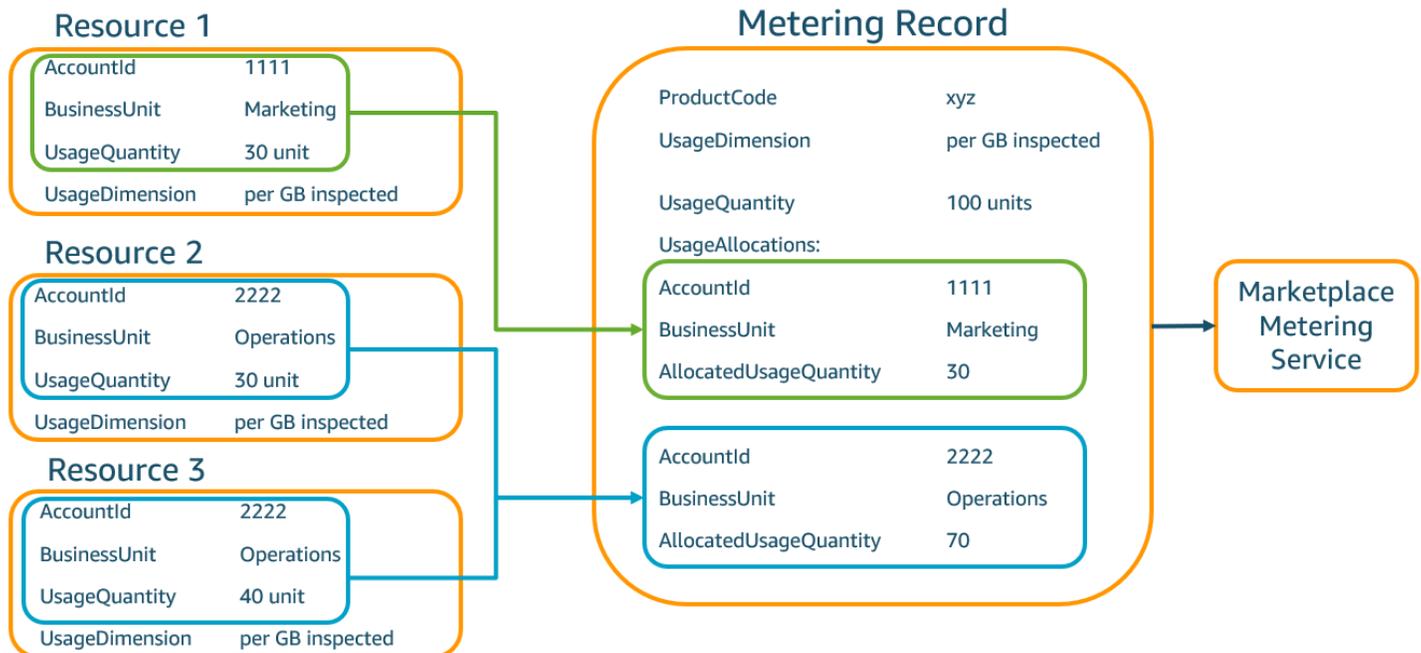
Dalam kasus penggunaan umum, pembeli berlangganan produk Anda dengan satu Akun AWS. Pembeli juga memiliki banyak pengguna yang terkait dengan langganan produk yang sama. Anda dapat membuat alokasi penggunaan dengan tag yang memiliki kunci `Account ID`, dan kemudian mengalokasikan penggunaan untuk setiap pengguna. Dalam hal ini, pembeli dapat mengaktifkan `Account ID` tag di konsol Billing and Cost Management mereka dan menganalisis penggunaan pengguna individu.

Pengalaman penjual

Penjual dapat menggabungkan catatan pengukuran untuk sumber daya dengan kumpulan tag yang sama alih-alih menggabungkan penggunaan untuk semua sumber daya. Misalnya, penjual dapat membuat catatan pengukuran yang mencakup ember yang berbeda. `UsageAllocations` Setiap bucket mewakili `UsageQuantity` satu set tag, seperti `AccountId` dan `BusinessUnit`.

Dalam diagram berikut, Resource 1 memiliki satu set `AccountId` dan `BusinessUnit` tag yang unik, dan muncul di Rekaman Pengukuran sebagai entri tunggal.

Resource 2 dan Resource 3 keduanya memiliki AccountId tag yang sama 2222,, dan BusinessUnit tag yang sama, Operations. Akibatnya, mereka digabungkan menjadi satu UsageAllocations entri dalam Rekaman Pengukuran.



Penjual juga dapat menggabungkan sumber daya tanpa tag menjadi satu UsageAllocation dan mengirimkannya sebagai salah satu entri di UsageAllocations.

Batasan meliputi:

- Jumlah tag — 5
- Ukuran UsageAllocations (kardinalitas) — 2.500
- Ukuran permintaan maksimal - 1 MB

Validasi meliputi:

- Karakter diperbolehkan untuk kunci tag dan nilai — a-zA-Z 0-9+ -= . _:\ /@
- Tag maksimum di seluruh UsageAllocation daftar - 5
- Dua tidak UsageAllocations dapat memiliki tag yang sama (yaitu, kombinasi yang sama dari kunci tag dan nilai). Jika itu masalahnya, mereka harus menggunakan hal yang sama UsageAllocation.
- Jumlah AllocatedUsageQuantity UsageAllocation harus sama dengan UsageQuantity, yang merupakan penggunaan agregat.

- Ukuran muatan maksimum tidak boleh lebih dari 1 MB. Ini termasuk kunci atribut input (misalnya, `UsageRecords`, `AllocatedUsageQuantity`, `tag`).

Note

Untuk memastikan bahwa Anda tidak melanggar batas payload, buat objek permintaan sampel dengan ukuran maksimum berdasarkan kebutuhan bisnis, ubah objek menjadi string JSON, dan dapatkan ukuran dalam byte. Pastikan bahwa satu panggilan API tidak akan melanggar batas 1 MB. Sebagai contoh, jika permintaan dengan 1 `UsageRecord` memiliki ukuran maksimum 200 KB, jangan mengirim lebih dari 5 `UsageRecords` sebagai bagian dari permintaan ($200\text{KB} * 5 = 1\text{MB}$).

Pengalaman pembeli

Tabel berikut menunjukkan contoh pengalaman pembeli setelah pembeli mengaktifkan tag `AccountId` dan `BusinessUnit` vendor.

Dalam contoh ini, pembeli dapat melihat penggunaan yang dialokasikan dalam Laporan Penggunaan Biaya mereka. Tag yang diukur vendor menggunakan awalan. “`aws:marketplace:isv`” Pembeli dapat mengaktifkannya di Billing and Cost Management, di bawah Tag Alokasi Biaya AWS, tag alokasi biaya yang dihasilkan.

Baris pertama dan terakhir dari Laporan Penggunaan Biaya relevan dengan apa yang dikirim Penjual ke Layanan Pengukuran (seperti yang ditunjukkan dalam [Pengalaman penjual](#) contoh).

Laporan Penggunaan Biaya (Sederhana)

ProductCode	Pembeli	UsageDimension	UsageQuantity	<code>aws:marketplace:isv:AccountId</code>	<code>aws:marketplace:isv:BusinessUnit</code>
xyz	11112222333	Jaringan: per (GB) diperiksa	70	2222	Operasi

ProductCode	Pembeli	UsageDimension	UsageQuantity	aws:marketplace:isv:AccountId	aws:marketplace:isv:BusinessUnit
xyz	111122223333	Jaringan: per (GB) diperiksa	30	3333	Keuangan
xyz	111122223333	Jaringan: per (GB) diperiksa	20	4444	IA
xyz	111122223333	Jaringan: per (GB) diperiksa	20	5555	Pemasaran
xyz	111122223333	Jaringan: per (GB) diperiksa	30	1111	Pemasaran

Untuk contoh kode, lihat [BatchMeterUsagedengan contoh kode penandaan alokasi penggunaan \(Opsional\)](#).

Memeriksa hak menggunakan AWS Marketplace Entitlement Service

Jika produk Anda adalah produk kontrak SaaS, produk Anda memanggil AWS Marketplace Entitlement Service untuk mengambil hak pelanggan menggunakan API. [GetEntitlements](#) Produk Anda harus memverifikasi penggunaan selanjutnya pada akun tersebut terhadap Layanan AWS Marketplace Hak. Misalnya, jika pelanggan memberikan 10 pengguna pada akun, produk Anda harus memeriksa Layanan AWS Marketplace Hak untuk hak atas kapasitas tersebut.

Untuk memverifikasi hak pelanggan atas produk Anda, gunakan `GetEntitlements` operasi di Layanan AWS Marketplace Hak. Layanan AWS Marketplace Hak hanya tersedia di Wilayah AS Timur (Virginia N.), dapat diakses melalui `entitlement.marketplace.us-east-1.amazonaws.com` atau `entitlement-marketplace.us-east-1.api.aws`

`GetEntitlements` menerima ID akun AWS pelanggan, pengenalan pelanggan, dan dimensi sebagai filter. `ProductCode` adalah parameter yang diperlukan. Operasi mengembalikan daftar hak paginasi. Hasilnya memiliki `ExpirationDate` bidang yang menunjukkan periode waktu minimum yang sah untuk hak tersebut. Jika pelanggan telah mengatur perpanjangan otomatis, tanggal di `ExpirationDate` lapangan adalah tanggal perpanjangan.

Untuk contoh kode, lihat [Contoh kode untuk integrasi produk SaaS](#).

Mengambil hak atas tindakan pengguna

Contoh berikut dapat membantu Anda lebih memahami proses pengambilan hak atas tindakan pengguna.

Contoh: Produk berbasis pengguna

Anda menawarkan produk yang memungkinkan sejumlah akun ada untuk pelanggan tertentu. Pelanggan dapat mengunjungi dasbor untuk menyediakan pengguna baru (misalnya, untuk menetapkan kredensi). Ketika pelanggan memberikan pengguna baru, produk Anda menelepon `GetEntitlements` untuk memverifikasi bahwa kapasitas itu ada. Jika tidak, Anda dapat menghubungi Layanan AWS Marketplace Pengukuran untuk menagih pengguna tambahan.

Contoh: Produk penyimpanan data

Anda menawarkan produk yang memungkinkan pelanggan untuk menyimpan sejumlah data dalam bentuk terenkripsi atau tidak terenkripsi. Pelanggan dapat melihat dasbor yang menampilkan jumlah data yang ada dan dialokasikan dalam produk Anda. Dasbor Anda mengambil jumlah alokasi melalui `GetEntitlements`.

Daftar periksa integrasi produk SaaS

Sebelum produk perangkat lunak sebagai layanan (SaaS) Anda ditayangkan, Anda harus memverifikasi bahwa Anda telah menyelesaikan konfigurasi yang diperlukan. Anda dapat menggunakan daftar periksa integrasi SaaS berikut untuk memverifikasi bahwa Anda telah menyelesaikan konfigurasi yang diperlukan.

Kategori	Persyaratan
Akses	Mengirimkan formulir pendaftaran penjual dengan AWS akun yang diinginkan untuk AWS Marketplace digunakan.

Kategori	Persyaratan
Akses	Menyelesaikan pendaftaran penjual, termasuk syarat dan ketentuan, rekening bank, dan formulir pajak W8 atau W9.
Akses	Peran lintas akun yang dikonfigurasi untuk AWS Marketplace akun terdaftar.
Produk	Menyelesaikan formulir permintaan produk di Portal Manajemen AWS Marketplace.
Produk	AWS Akun yang disediakan IDs untuk pengujian di tab Catatan dari wizard Buat produk di AMMP.
Produk	Menyediakan URL EULA dalam format.txt di tab Produk.
Produk	Menerima kode produk Anda dan informasi topik Amazon SNS dari AWS Marketplace
Produk	Berlangganan topik Amazon SNS dan membuat antrian Amazon SQS untuk berlangganan topik Amazon SNS.
Solusi Penagihan	Divalidasi Anda dapat mengirim catatan pengukuran ke <code>BatchMeterUsage</code> operasi setiap jam untuk setiap pelanggan untuk produk langganan SaaS. Dapat mengirim catatan pengukuran untuk penggunaan tambahan oleh setiap pelanggan untuk produk kontrak SaaS.
Solusi Penagihan	Divalidasi Anda dapat memverifikasi hak pelanggan dari AWS Marketplace Layanan Hak untuk produk kontrak SaaS.

Kategori	Persyaratan
Solusi Penagihan	Divalidasi bahwa biaya muncul seperti yang diharapkan pada tagihan yang dihasilkan untuk akun pengujian.
Solusi Penagihan	Diuji untuk situasi seperti pelanggan tidak valid IDs dan langganan yang dibatalkan.
Produk	Mengirimkan permintaan produk kembali ke AWS Marketplace penerbitan.
Registrasi	Menerapkan halaman pendaftaran HTTPS yang dapat menerima permintaan HTTP POST.
Registrasi	Divalidasi Anda dapat menerima pendaftaran pelanggan baru.
Registrasi	Divalidasi Anda tidak menyimpan token pendaftaran dalam cookie.
Registrasi	Divalidasi yang Anda gunakan <code>ResolveCustomer</code> untuk mendapatkan <code>ProductCode</code> dan <code>CustomerIdentifier</code> dari AWS token.
Registrasi	Divalidasi Anda dapat menyelesaikan token pendaftaran yang diterima dari AWS tanpa penundaan.
Registrasi	Diuji bahwa Anda tidak diblokir untuk mendaftar dengan alamat layanan email seperti Gmail.
Registrasi	Diuji bahwa Anda dapat menerima pendaftaran yang tidak lengkap dan beberapa upaya pendaftaran.

Kategori	Persyaratan
Langganan	Uji yang dapat Anda tangani unsubscribe-pending dan unsubscribe-success pesan.
Langganan	Divalidasi bahwa Anda mengirim catatan pengukuran akhir dalam waktu satu jam setelah menerima pesan. unsubscribe-pending
Keamanan	Akun AWS root yang divalidasi tidak memiliki kunci API, memiliki kata sandi yang kuat, dan dikaitkan dengan perangkat otentikasi multi-faktor perangkat keras (MFA). Semua akses administratif adalah melalui identitas yang dibuat dengan AWS Identity and Access Management (IAM). Tidak ada akun bersama.
Keamanan	Memvalidasi bahwa peran IAM digunakan untuk semua akses Amazon Elastic Compute Cloud (Amazon EC2) terprogram. Kredensial tidak dikodekan ke dalam skrip, header, atau kode sumber.
Keamanan	Divalidasi Anda mempertahankan pencatatan dan konsolidasi log yang komprehensif.
Keamanan	Terverifikasi bahwa Anda memiliki batas subnet publik dan pribadi yang didefinisikan dengan baik yang mengisolasi layanan aplikasi dan akses ke database dan sistem file. Definisi kelas data berbeda yang membatasi data sensitif dan memisahkan data publik dan pribadi.
Keamanan	Diverifikasi bahwa Anda memiliki enkripsi data pribadi dalam perjalanan dan saat istirahat dengan rotasi kunci terjadwal.

Kategori	Persyaratan
Keamanan	Divalidasi Anda memiliki alat insiden keamanan dan akses di tempat dan latihan respons insiden yang dijadwalkan secara rutin yang mengakomodasi penyelidikan dan pemulihan tepat waktu.
Keandalan	Terverifikasi sistem beradaptasi dengan perubahan permintaan, meningkatkan dan menurunkan skala sesuai kebutuhan, dan menggunakan penyeimbangan beban untuk memastikan kinerja tinggi. Sistem ini juga menyediakan caching berbasis tepi sesuai kebutuhan.
Keandalan	Waktu pemulihan dan tujuan titik yang divalidasi ditentukan, dan pemulihan bencana dijadwalkan secara berkala. Kegagalan komponen adalah penyembuhan diri melalui pemicu dan pemberitahuan otomatis.

Pelaporan untuk produk SaaS di AWS Marketplace

AWS Marketplace menghasilkan laporan untuk perangkat lunak Anda sebagai produk layanan (SaaS) yang mencakup data pelanggan, keuangan, penggunaan, dan pajak. Tabel berikut memberikan informasi tentang laporan yang tersedia. Anda dapat menggunakan laporan untuk mendapatkan lebih banyak wawasan tentang basis pelanggan Anda dan lebih memahami keuangan Anda, termasuk penjualan dan pajak. Untuk informasi selengkapnya, lihat [the section called “Laporan penjual”](#). Tabel berikut merangkum bagaimana keuangan untuk produk SaaS dilaporkan.

Laporan	Konten SaaS
Laporan bisnis harian	Biaya kontrak di muka muncul di bagian Biaya.

Laporan	Konten SaaS
	Biaya penggunaan terukur muncul di bagian Penggunaan.
Laporan pendapatan bulanan	Biaya kontrak di muka muncul di bagian Langganan tahunan. Biaya penggunaan terukur muncul di bagian Data penagihan dan pendapatan.
Laporan kompensasi penjualan	Biaya kontrak di muka dan biaya penggunaan tambahan bulanan muncul sebagai item baris terpisah.
Laporan pelanggan pelanggan	Kontrak SaaS baru muncul di bagian langganan Tahunan. Langganan SaaS baru muncul di bagian langganan per jam/bulanan.

Contoh kode untuk integrasi produk SaaS

Anda dapat menggunakan contoh kode berikut untuk mengintegrasikan perangkat lunak Anda sebagai produk layanan (SaaS) dengan AWS Marketplace APIs yang diperlukan untuk menerbitkan dan memelihara produk Anda. Untuk informasi selengkapnya, silakan lihat bagian-bagian berikut ini.

Topik

- [ResolveCustomercontoh kode](#)
- [GetEntitlementcontoh kode](#)
- [BatchMeterUsagecontoh kode](#)
- [BatchMeterUsagedengan contoh kode penandaan alokasi penggunaan \(Opsional\)](#)

ResolveCustomercontoh kode

Contoh kode berikut relevan untuk semua model penetapan harga. Contoh Python menukar `x-amzn-marketplace-token` token untuk `CustomerIdentifier`, `ProductCode`, dan.

`CustomerAWSAccountId` `CustomerAWSAccountId` ini adalah Akun AWS ID yang terkait dengan langganan. Kode ini berjalan dalam aplikasi di situs web pendaftaran Anda, ketika Anda dialihkan ke sana dari Portal Manajemen AWS Marketplace. Pengalihan adalah permintaan POST yang menyertakan token.

Untuk informasi selengkapnya `ResolveCustomer`, lihat [ResolveCustomer](#) di Referensi AWS Marketplace API Layanan Pengukuran.

```
# Import AWS Python SDK and urllib.parse
import boto3
import urllib.parse as urlparse

# Resolving Customer Registration Token
formFields = urlparse.parse_qs(postBody)
regToken = formFields['x-amzn-marketplace-token'][0]

# If regToken present in POST request, exchange for customerID
if (regToken):
    marketplaceClient = boto3.client('meteringmarketplace')
    customerData = marketplaceClient.resolve_customer(RegistrationToken=regToken)
    productCode = customerData['ProductCode']
    customerID = customerData['CustomerIdentifier']
    customerAWSAccountId = customerData['CustomerAWSAccountId']

# TODO: Store customer information
# TODO: Validate no other accounts share the same customerID
```

Contoh tanggapan

```
{
  'CustomerIdentifier': 'string',
  'CustomerAWSAccountId': 'string',
  'ProductCode': 'string'
}
```

GetEntitlementcontoh kode

Contoh kode berikut relevan untuk produk SaaS dengan kontrak dan kontrak SaaS dengan model harga konsumsi. Contoh Python memverifikasi bahwa pelanggan memiliki hak aktif.

Untuk informasi selengkapnya `GetEntitlement`, lihat [GetEntitlement](#) di Referensi API Layanan AWS Marketplace Hak.

```
# Import AWS Python SDK
import boto3

marketplaceClient = boto3.client('marketplace-entitlement', region_name='us-east-1')

# Filter entitlements for a specific customerID
#
# productCode is supplied after the AWS Marketplace Ops team has published
# the product to limited
#
# customerID is obtained from the ResolveCustomer response
entitlement = marketplaceClient.get_entitlements({
    'ProductCode': 'productCode',
    'Filter' : {
        # Option 1: Using CustomerIdentifier (deprecated after Dec 31, 2025)
        'CUSTOMER_IDENTIFIER': [
            'customerID',
        ]
        # Option 2: Using CustomerAWSAccountID (preferred)
        # 'CUSTOMER_AWS_ACCOUNT_ID': [
        #     'awsAccountID',
        # ]
    },
    'NextToken' : 'string',
    'MaxResults': 123
})

# TODO: Verify the dimension a customer is subscribed to and the quantity,
# if applicable
```

Contoh tanggapan

Nilai yang dikembalikan sesuai dengan dimensi yang dibuat saat Anda membuat produk di Portal Manajemen AWS Marketplace.

```
{
  "Entitlements": [
    {
      "CustomerIdentifier": "string",
      "CustomerAWSAccountID": "string",
```

```

    "Dimension": "string",
    "ExpirationDate": number,
    "ProductCode": "string",
    "Value": {
      "BooleanValue": boolean,
      "DoubleValue": number,
      "IntegerValue": number,
      "StringValue": "string"
    }
  }
],
"NextToken": "string"
}

```

BatchMeterUsagecontoh kode

Contoh kode berikut relevan untuk langganan SaaS dan kontrak dengan model harga konsumsi, tetapi tidak untuk produk kontrak SaaS tanpa konsumsi. Contoh Python mengirimkan catatan pengukuran untuk membebankan biaya AWS Marketplace kepada pelanggan Anda. pay-as-you-go

```

# NOTE: Your application will need to aggregate usage for the
#       customer for the hour and set the quantity as seen below.
# AWS Marketplace can only accept records for up to an hour in the past.
#
# productCode is supplied after the AWS Marketplace Ops team has
# published the product to limited
#
# You can use either:
# - customerID from the ResolveCustomer response (deprecated after Dec 31, 2025)
# - AWS account ID of the buyer

# Import AWS Python SDK
import boto3
from datetime import datetime

# Option 1: Using CustomerIdentifier (deprecated after Dec 31, 2025)
usageRecord = [
    {
        'Timestamp': datetime(2015, 1, 1),
        'CustomerIdentifier': 'customerID',
        'Dimension': 'string',
        'Quantity': 123
    }
]

```

```

]

# Option 2: Using CustomerAWSAccountID (preferred)
# usageRecord = [
#     {
#         'Timestamp': datetime(2015, 1, 1),
#         'CustomerAWSAccountID': 'awsAccountID',
#         'Dimension': 'string',
#         'Quantity': 123
#     }
# ]

marketplaceClient = boto3.client('meteringmarketplace')

response = marketplaceClient.batch_meter_usage(
    UsageRecords=usageRecord,
    ProductCode='productCode'
)

```

Untuk informasi selengkapnya `BatchMeterUsage`, lihat [BatchMeterUsage](#) di Referensi AWS Marketplace API Layanan Pengukuran.

Contoh tanggapan

```

{
  'Results': [
    {
      'UsageRecord': {
        'Timestamp': datetime(2015, 1, 1),
        'CustomerIdentifier': 'string',
        'CustomerAWSAccountID': 'string',
        'Dimension': 'string',
        'Quantity': 123
      },
      'MeteringRecordId': 'string',
      'Status': 'Success' | 'CustomerNotSubscribed' | 'DuplicateRecord'
    },
  ],
  'UnprocessedRecords': [
    {
      'Timestamp': datetime(2015, 1, 1),
      'CustomerIdentifier': 'string',
      'CustomerAWSAccountID': 'string',

```

```

        'Dimension': 'string',
        'Quantity': 123
    }
]
}

```

BatchMeterUsagedengan contoh kode penandaan alokasi penggunaan (Opsional)

Contoh kode berikut relevan untuk langganan SaaS dan kontrak dengan model harga penggunaan, tetapi tidak untuk produk kontrak SaaS tanpa penggunaan. Contoh Python mengirimkan catatan pengukuran dengan tag alokasi penggunaan yang sesuai untuk membebaskan biaya AWS Marketplace kepada pelanggan Anda. pay-as-you-go

```

# NOTE: Your application will need to aggregate usage for the
#       customer for the hour and set the quantity as seen below.
# AWS Marketplace can only accept records for up to an hour in the past.
#
# productCode is supplied after the AWS Marketplace Ops team has
# published the product to limited
#
# You can use either:
# - customerID from the ResolveCustomer response (deprecated after Dec 31, 2025)
# - AWS account ID of the buyer

# Import AWS Python SDK
import boto3
import time

# Option 1: Using CustomerIdentifier (deprecated after Dec 31, 2025)
usageRecords = [
    {
        "Timestamp": int(time.time()),
        "CustomerIdentifier": "customerID",
        "Dimension": "Dimension1",
        "Quantity": 3,
        "UsageAllocations": [
            {
                "AllocatedUsageQuantity": 2,
                "Tags": [
                    { "Key": "BusinessUnit", "Value": "IT" },
                    { "Key": "AccountId", "Value": "*****" },
                ]
            }
        ]
    }
]

```

```

    ]
    },
    {
        "AllocatedUsageQuantity": 1,
        "Tags": [
            { "Key": "BusinessUnit", "Value": "Finance" },
            { "Key": "AccountId", "Value": "*****" },
        ]
    },
]
}
]

# Option 2: Using CustomerAWSAccountID (preferred)
# usageRecords = [
#     {
#         "Timestamp": int(time.time()),
#         "CustomerAWSAccountID": "awsAccountID",
#         "Dimension": "Dimension1",
#         "Quantity": 3,
#         "UsageAllocations": [
#             {
#                 "AllocatedUsageQuantity": 2,
#                 "Tags": [
#                     { "Key": "BusinessUnit", "Value": "IT" },
#                     { "Key": "AccountId", "Value": "*****" },
#                 ]
#             },
#             {
#                 "AllocatedUsageQuantity": 1,
#                 "Tags": [
#                     { "Key": "BusinessUnit", "Value": "Finance" },
#                     { "Key": "AccountId", "Value": "*****" },
#                 ]
#             },
#         ]
#     }
# ]

marketplaceClient = boto3.client('meteringmarketplace')

response = marketplaceClient.batch_meter_usage(
    UsageRecords=usageRecords,
    ProductCode="testProduct"

```

)

Untuk informasi selengkapnya `BatchMeterUsage`, lihat [BatchMeterUsage](#) di Referensi AWS Marketplace Metering Service API.

Contoh tanggapan

```
{
  "Results": [
    {
      "Timestamp": "1634691015",
      "CustomerIdentifier": "customerID",
      "CustomerAWSAccountID": "awsAccountID",
      "Dimension": "Dimension1",
      "Quantity": 3,
      "UsageAllocations": [
        {
          "AllocatedUsageQuantity": 2,
          "Tags": [
            { "Key": "BusinessUnit", "Value": "IT" },
            { "Key": "AccountId", "Value": "*****" }
          ]
        },
        {
          "AllocatedUsageQuantity": 1,
          "Tags": [
            { "Key": "BusinessUnit", "Value": "Finance" },
            { "Key": "AccountId", "Value": "*****" }
          ]
        }
      ],
      "MeteringRecordId": "8fjef98ejf",
      "Status": "Success"
    }
  ],
  "UnprocessedRecords": [
    {
      "Timestamp": "1634691015",
      "CustomerIdentifier": "customerID",
      "CustomerAWSAccountID": "awsAccountID",
      "Dimension": "Dimension1",
      "Quantity": 3,
      "UsageAllocations": []
    }
  ]
}
```

```
}  
  ]  
}
```

Mengirimkan produk Anda melalui VPC Amazon menggunakan AWS PrivateLink

AWS Marketplace mendukung AWS PrivateLink, Layanan AWS yang memungkinkan Anda menggunakan jaringan Amazon untuk memberi pembeli akses ke produk yang Anda jual AWS Marketplace. Bagian berikut menguraikan proses untuk mengonfigurasi dan mengirimkan produk Anda melalui titik akhir Amazon Virtual Private Cloud (VPC) Amazon Virtual Cloud (VPC) menggunakan teknologi. AWS PrivateLink

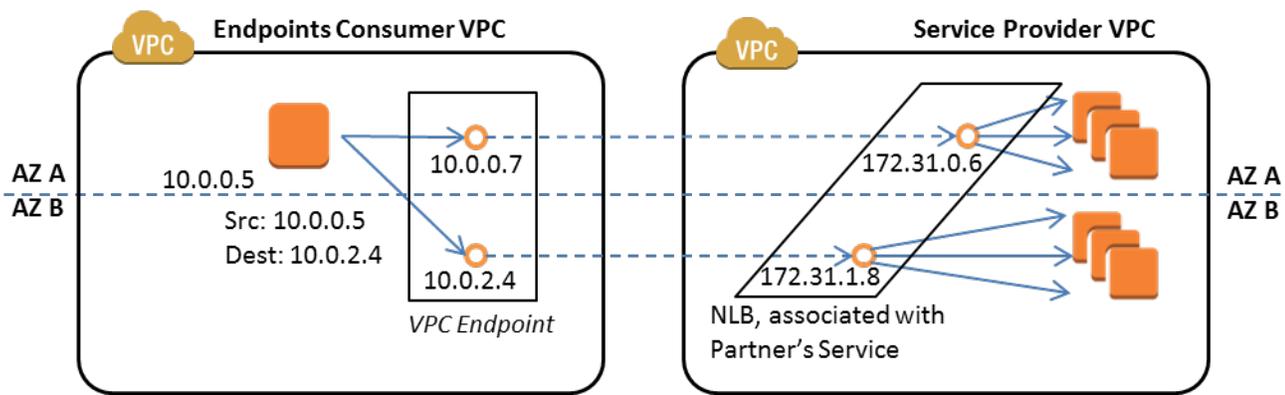
Topik ini mengasumsikan bahwa Anda memiliki pengetahuan tentang beberapa AWS layanan dan AWS Marketplace lingkungan.

Topik

- [Pengantar](#)
- [Mengkonfigurasi produk Anda](#)
- [Mengirimkan produk Anda ke AWS Marketplace](#)
- [Akses pembeli ke titik akhir VPC](#)
- [Lampiran: Daftar Periksa](#)

Pengantar

Sebagai AWS Marketplace penjual, Anda dapat memberikan akses kepada pembeli ke layanan Anda melalui titik akhir VPC Amazon. Pendekatan ini memberi pembeli akses ke layanan Anda di seluruh jaringan Amazon menggunakan [AWS PrivateLink](#) teknologi. Jika Anda menggunakan AWS Marketplace untuk membuat dan mengirimkan penawaran ini, pembeli dapat menemukan layanan Anda di AWS Marketplace. Pembeli Anda juga dapat menemukan produk Anda dalam daftar layanan yang tersedia untuk membuat titik akhir VPC.



[Endpoint VPC](#) adalah perangkat virtual yang memungkinkan AWS pelanggan untuk membuat koneksi pribadi antara VPC mereka dan AWS layanan lain tanpa memerlukan akses melalui internet, melalui perangkat NAT, koneksi VPN, atau AWS Direct Connect Anda dapat membuat layanan endpoint melalui AWS Marketplace yang memungkinkan pembeli untuk menggunakan teknologi ini untuk terhubung ke layanan Anda. Metode koneksi ini lebih aman bagi pembeli Anda karena mereka mengakses layanan Anda melalui jaringan pribadi Amazon daripada melalui Internet.

Untuk setiap wilayah tempat Anda ingin menawarkan layanan, Anda membuat atau menggunakan sumber daya yang ada untuk mengonfigurasi VPC, menyiapkan instance layanan, menyiapkan penyeimbang beban jaringan, dan mendaftarkan layanan Anda dengan penyeimbang beban jaringan dengan membuat titik akhir layanan. Setelah Anda menyelesaikan langkah-langkah tersebut dan menguji penawaran Anda, Anda memberikan informasi konfigurasi Anda kepada tim [Operasi AWS Marketplace Penjual](#).

AWS merekomendasikan agar Anda memberikan nama DNS pribadi yang dapat digunakan pembeli Anda saat mereka membuat titik akhir VPC.

Saat pembeli membuat titik akhir VPC mereka, mereka memiliki opsi untuk mengaktifkan nama DNS pribadi. Dengan memilih opsi ini, layanan VPC pembeli mengonfigurasi zona host [pribadi](#). Jika Anda memberikan nama DNS pribadi, pembeli dapat menggunakannya saat mengonfigurasi titik akhir VPC untuk terhubung ke layanan Anda. Di zona host pribadi pembeli, nama DNS pribadi (api.example.com) akan menunjuk ke nama DNS yang dihasilkan secara acak (vpce-11111111111111111111-yyy-yyyyy.api.vpce.example.com) yang dibuat untuk layanan titik akhir Anda. EC2 Instance pembeli memanggil nama DNS terpadu yang sama (api.example.com) di berbagai tempat. VPCs Selain itu, jika nama DNS publik dan pribadi sama, pembeli dapat menggunakan nama publik yang sama saat mengakses layanan Anda dari dalam atau di luar VPC.

Untuk bantuan dalam membuat layanan Anda tersedia AWS Marketplace, Anda dapat menghubungi tim [Operasi AWS Marketplace Penjual](#). Ketika AWS Marketplace pembeli berlangganan layanan

Anda dan membuat titik akhir VPC, layanan Anda ditampilkan di bawah Layanan Anda. AWS Marketplace Tim Operasi AWS Marketplace Penjual menggunakan nama DNS yang mudah digunakan untuk memudahkan penemuan layanan Anda saat membuat titik akhir VPC.

Produk Anda dibuat sebagai produk perangkat lunak sebagai layanan (SaaS). Pengukuran dan penagihan sama dengan produk SaaS lainnya AWS Marketplace .

Mengkonfigurasi produk Anda

Untuk mengonfigurasi produk Anda agar tersedia melalui titik akhir VPC Amazon:

1. Buat atau gunakan [Amazon VPC](#) yang sudah ada.
2. Buat (atau gunakan EC2 instans [Amazon](#) yang sudah ada) untuk produk Anda.
3. Buat [penyeimbang beban jaringan](#) di setiap wilayah tempat Anda menawarkan produk Anda. AWS merekomendasikan agar Anda menyertakan semua [Availability Zones](#) (AZs) untuk suatu wilayah.
4. Gunakan konsol VPC Amazon, CLI, atau didukung untuk SDKs membuat layanan titik akhir VPC.
5. Verifikasi bahwa Anda dapat mengakses layanan melalui penyeimbang beban jaringan.
6. [Minta sertifikat dari AWS Certificate Manager \(ACM\) untuk nama DNS](#) yang ramah pengguna Anda. Sebelum ACM mengeluarkan sertifikat, ACM memvalidasi bahwa Anda memiliki atau mengontrol nama domain dalam permintaan sertifikat Anda.
7. Delegasikan subdomain nama DNS yang mudah digunakan, seperti `api.vpce.example.com`, ke server nama yang diberikan kepada Anda oleh tim Operasi Penjual. AWS Marketplace Dalam sistem DNS Anda, Anda harus membuat catatan sumber daya server nama (NS) untuk mengarahkan subdomain ini ke server nama Amazon Route 53 yang disediakan oleh tim Operasi AWS Marketplace Penjual sehingga nama DNS (seperti `vpce-0ac6c347a78c90f8.api.vpce.example.com`) dapat diselesaikan secara publik.
8. Izinkan akses ke AWS akun pembeli Anda.

Catatan: Anda dapat menggunakan SDK yang didukung atau perintah CLI ini untuk mengotomatiskan akses ke akun: `aws modify-vpc-endpoint-service vpcev2 -permissions --service-id vpce-svc-0123456789abcdef1 -- arn:aws:iam: :1111111111:root arn:aws:iam: :2222222222:root. add-allowed-principals`

Mengirimkan produk Anda ke AWS Marketplace

Selama proses penerbitan layanan Anda AWS Marketplace, Anda bekerja dengan tim Operasi AWS Marketplace Penjual. Untuk mengirimkan produk PrivateLink berkemampuan Anda:

1. Email informasi berikut ke tim [Operasi AWS Marketplace Penjual](#):

- a. Endpoint dan AWS akun yang digunakan untuk membuat endpoint. Titik akhirnya mirip dengan ini: `com.amazonaws.vpce.us-east-1.vpce-svc-0daa010345a21646`
- b. Nama DNS yang mudah digunakan untuk layanan Anda. Ini adalah nama DNS yang digunakan AWS Marketplace pembeli untuk mengakses produk Anda.
- c. AWS Akun yang Anda gunakan untuk meminta sertifikat dan pembeli nama DNS pribadi digunakan untuk mengakses titik akhir VPC.

Tim Operasi AWS Marketplace Penjual memverifikasi identitas perusahaan Anda dan nama DNS yang akan digunakan untuk layanan yang Anda daftarkan (seperti `api.vpce.example.com`). Setelah verifikasi, nama DNS akan mengganti nama DNS titik akhir dasar default.

Akses pembeli ke titik akhir VPC

AWS Marketplace pembeli yang membuat titik akhir VPC dapat menemukan layanan Anda dalam situasi ini:

- Anda mengikuti proses penjual yang dijelaskan sebelumnya di halaman ini untuk membuat atau menggunakan produk yang sudah ada.
- Pembeli berlangganan layanan Anda.
- Anda menambahkan AWS akun pembeli ke daftar akun yang diizinkan.

Saat pembeli membuat titik akhir VPC, mereka memiliki opsi untuk mengaitkan zona host pribadi dengan VPC mereka. Zona yang dihosting berisi kumpulan catatan untuk nama DNS pribadi default untuk layanan yang menyelesaikan ke alamat IP pribadi antarmuka jaringan titik akhir di VPC mereka.

Setiap titik akhir yang dihosting pembeli, termasuk AWS Marketplace layanan, dapat memberikan izin ke semua akun (izin `“*”`). Namun, ketika Anda menggunakan pendekatan ini, layanan tidak disertakan dalam panggilan Deskripsi atau konsol kecuali Anda mencari berdasarkan nama layanan. Untuk menampilkan layanan dalam panggilan Deskripsi, AWS akun pembeli harus ditambahkan secara eksplisit ke daftar izin oleh layanan.

Untuk mengakses layanan Anda, pembeli melakukan hal berikut:

1. Temukan dan berlangganan layanan Anda di AWS Marketplace.

2. Gunakan AWS Command Line Interface (AWS CLI), API, atau konsol VPC Amazon untuk menemukan layanan Anda dan kemudian buat titik akhir VPC untuk terhubung ke layanan Anda di subnet dan mereka gunakan. AZs Titik akhir ditampilkan sebagai antarmuka jaringan elastis di subnet. Alamat IP lokal dan nama DNS wilayah dan zona ditetapkan ke titik akhir.

Nama DNS sisi klien	Nama
Regional	<0dc9a211a78c90f8>Vpce .api.vpce.example.com
IAD2 (1a)	us-timur-1a <0dc9a211a78c90f8>-Vpce .api.vpce.example.com
IAD2 (1b)	us-timur-1b -Vpce <0dc9a211a78c90f8>.api.vpce.example.com

Jika Anda memberikan nama DNS pribadi default dan pembeli memilih Aktifkan Nama DNS Pribadi (terkait zona host pribadi) saat membuat titik akhir VPC, pembeli akan melihat nama DNS pribadi default regional untuk terhubung ke layanan Anda.

Nama	Alias	ID zona yang dihosting alias	(Catatan)
api.example.com	<0dc9a211a78c90f8>vpce. api.vpce.example.com	Z00AABBCCDD	IAD1 IAD2

Lampiran: Daftar Periksa

Gunakan daftar periksa berikut untuk memastikan bahwa Anda mengonfigurasi dan menguji produk Anda sebelum mengirimkannya ke tim Operasi AWS Marketplace Penjual.

Daftar periksa pembuatan produk

- Buat (atau gunakan VPC yang sudah ada) dan kemudian konfigurasi.
- Buat dan konfigurasi penyeimbang beban jaringan dalam VPC.

- Daftarkan layanan Anda dengan penyeimbang beban jaringan Anda dengan membuat layanan titik akhir VPC.
- Berikan ID AWS akun yang Anda gunakan untuk mengonfigurasi titik akhir VPC ke tim Operasi AWS Marketplace Penjual.
- Berikan nama layanan titik akhir default (misalnya, com.amazonaws.vpce.us-east-1.vpce-svc-0bbb070044a2164) ke tim Operasi Penjual. AWS Marketplace
- Berikan nama DNS layanan yang mudah digunakan (wajib) untuk mengganti nama DNS layanan yang dibuat secara acak. Minta sertifikat SSL dari ACM untuk subdomain yang digunakan untuk nama DNS layanan ramah pengguna Anda. Berikan sertifikat ini dan ID AWS akun yang Anda gunakan untuk memintanya ke tim Operasi AWS Marketplace Penjual.
- Direkomendasikan: Berikan nama DNS pribadi.
- Buat proses untuk menginformasikan dan memungkinkan AWS Marketplace pembeli Anda opsi untuk terhubung ke layanan Anda menggunakan AWS PrivateLink teknologi. Tambahkan AWS akun IDs untuk pembeli Anda ke daftar akun yang diizinkan.

Pengujian produk

- Verifikasi bahwa layanan Anda dikonfigurasi dan dapat ditemukan.
- Verifikasi bahwa layanan Anda dapat ditemukan melalui penyeimbang beban jaringan.
- Verifikasi bahwa pembeli dapat membuat titik akhir VPC dan mengakses layanan Anda. Gunakan AWS akun yang Anda miliki yang bukan akun yang Anda gunakan untuk mengatur layanan Anda.

Produk layanan profesional di AWS Marketplace

Sebagai penjual, Anda dapat menawarkan layanan profesional kepada AWS Marketplace pembeli. Layanan profesional mencakup layanan untuk menilai, bermigrasi, mendukung, mengelola, dan melatih orang lain tentang cara menggunakan AWS layanan dan produk di AWS Marketplace. Penjual membuat penawaran produk yang menggambarkan layanan yang mereka berikan, bernegosiasi dengan pelanggan untuk membuat perjanjian tentang persyaratan, dan kemudian membuat penawaran khusus untuk layanan melalui AWS Marketplace.

Note

Sebagai vendor perangkat lunak independen (ISV), mitra saluran, atau mitra konsultasi, Anda juga dapat memberi wewenang kepada mitra lain untuk menjual kembali produk layanan profesional Anda menggunakan penawaran pribadi mitra saluran. Untuk informasi selengkapnya, lihat [Membuat peluang jual kembali untuk mitra saluran sebagai ISV](#).

Pembeli dapat menemukan produk layanan profesional di AWS Marketplace katalog dengan memilih Layanan Profesional di bawah Kategori, memilih Layanan Profesional di bawah metode Pengiriman, dan menyempurnakan pencarian mereka berdasarkan Penerbit, model Harga, dan unit Harga. Mereka dikenakan biaya untuk layanan dalam AWS tagihan mereka. Mereka dapat menggunakan alat-alat seperti AWS Cost Explorer untuk memusatkan pembayaran dan mengelola biaya mereka.

Untuk informasi lebih lanjut tentang produk layanan profesional, lihat:

- [Memulai dengan produk layanan profesional](#)
- [Memberikan rincian untuk produk layanan profesional](#)
- [Persyaratan untuk produk layanan profesional](#)
- [Menciptakan peluang jual kembali untuk mitra saluran sebagai ISV](#)

Video berikut menjelaskan lebih lanjut tentang mengelola produk layanan profesional di AWS Marketplace.

[Kelola produk layanan Profesional di AWS Marketplace.](#)

Mendapatkan bantuan

Untuk bantuan dengan produk layanan profesional Anda, hubungi mitra pengembangan bisnis Anda untuk AWS Marketplace atau tim [Operasi AWS Marketplace Penjual](#).

Memulai dengan produk layanan profesional di AWS Marketplace

Sebagai penjual, Anda dapat menawarkan layanan profesional kepada AWS Marketplace pembeli. Layanan profesional mencakup layanan untuk menilai, bermigrasi, mendukung, mengelola, dan melatih orang lain tentang cara menggunakan AWS layanan dan produk di AWS Marketplace. Definisi produk Anda memberi tahu pembeli tentang layanan yang Anda tawarkan dan menjelaskan mengapa mereka harus memilih perusahaan Anda untuk layanan tersebut. AWS Marketplace Memungkinkan pembeli untuk menghubungi Anda. Anda menyetujui kontrak, dan kemudian Anda membuat penawaran pribadi yang memungkinkan pembeli membeli layanan Anda dengan biaya tetap. Bagian berikut menunjukkan kepada Anda bagaimana memulai dengan produk layanan profesional, memberikan langkah-langkah yang Anda butuhkan untuk membuat produk pertama Anda dan menawarkannya kepada pelanggan Anda.

Video berikut menjelaskan lebih lanjut tentang daftar produk layanan profesional di AWS Marketplace.

Topik

- [Prasyarat](#)
- [Buat produk layanan profesional](#)
- [Buat penawaran pribadi](#)
- [Edit informasi produk](#)
- [Mengedit harga produk](#)
- [Edit visibilitas produk](#)
- [Hapus produk layanan profesional](#)

Prasyarat

Untuk menjual layanan profesional AWS Marketplace, Anda harus menyelesaikan prasyarat berikut:

- Memiliki akses ke Portal Manajemen AWS Marketplace. Ini adalah alat yang Anda gunakan untuk mendaftar sebagai penjual dan mengelola produk yang Anda jual AWS Marketplace. Untuk

mempelajari lebih lanjut tentang mendapatkan akses ke Portal Manajemen AWS Marketplace, lihat [Kebijakan dan izin untuk penjual AWS Marketplace](#).

- Daftar sebagai AWS Marketplace penjual dan kirimkan informasi pajak dan perbankan Anda. Anda harus menyediakan rekening bank non-Inggris untuk menerima pencairan. Untuk mempelajari lebih lanjut tentang menjadi penjual, lihat [Memulai sebagai AWS Marketplace penjual](#).
- Anda harus memiliki produk layanan profesional untuk ditawarkan yang terkait dengan AWS layanan atau setidaknya satu produk publik di AWS Marketplace. Produk Anda harus secara langsung mendukung produk tersebut, atau menawarkan layanan yang mendorong langganan produk tersebut.

Note

Produk Anda harus terdaftar dalam setidaknya satu dari kategori utama ini: Penilaian, Implementasi, Layanan terkelola, dukungan Premium, atau Pelatihan. Untuk informasi selengkapnya tentang pedoman produk layanan profesional, lihat [Persyaratan untuk produk layanan profesional AWS Marketplace](#).

Buat produk layanan profesional

Prosedur berikut menjelaskan cara membuat produk layanan profesional baru di Portal Manajemen AWS Marketplace.

Untuk membuat produk layanan profesional

1. Buka browser web dan masuk ke file [Portal Manajemen AWS Marketplace](#).
2. Dari menu Produk, pilih Layanan profesional. Halaman ini menunjukkan kepada Anda semua produk layanan profesional yang telah Anda buat, serta permintaan apa pun yang Anda buat untuk membuat atau memodifikasi produk ini.
3. Pada tab Produk Layanan Profesional, pilih Buat produk layanan profesional.
4. Pada halaman Buat produk, berikan informasi untuk produk Anda, dan pilih Kirim. Untuk informasi lebih lanjut tentang detail yang harus Anda berikan, lihat [Memberikan rincian untuk produk layanan profesional di AWS Marketplace](#).
5. (Opsional) Dari menu Produk Portal Manajemen AWS Marketplace, pilih Layanan profesional, lalu pilih tab Permintaan. Verifikasi bahwa Anda melihat permintaan produk Anda dengan judul

Produk yang benar, dan bahwa status Permintaan sedang ditinjau. Produk Anda harus dibuat dalam mode pratinjau terbatas dalam beberapa menit.

Note

Anda dapat kembali ke tab Permintaan di halaman Layanan Profesional untuk melihat status permintaan Anda kapan saja. Setiap kesalahan dalam proses pembuatan akan muncul di sini. Anda dapat memilih permintaan untuk melihat detail permintaan atau untuk memperbaiki kesalahan.

Ketika produk Anda awalnya dibuat, itu hanya dapat diakses oleh Anda Akun AWS (yang Anda gunakan untuk membuat produk). Jika Anda melihat produk dari halaman Layanan profesional, Anda dapat memilih Lihat AWS Marketplace untuk melihat detail produk saat muncul AWS Marketplace untuk pembeli. Daftar detail ini tidak tersedia untuk AWS Marketplace pengguna lain, kecuali jika Anda memberikan penawaran pribadi kepada mereka.

Untuk mempelajari cara membuat produk tersedia untuk umum, lihat [Edit visibilitas produk](#).

Buat penawaran pribadi

Ketika pembeli potensial melihat produk Anda AWS Marketplace, mereka tidak dapat membelinya secara langsung. Ketika mereka mencoba untuk berlangganan, mereka diarahkan untuk meminta [penawaran pribadi](#) dari Anda. AWS Marketplace mengirim pesan email ke alamat email pengguna root akun AWS Marketplace penjual Anda, memberi tahu Anda bahwa pelanggan telah meminta penawaran pribadi. Prosedur berikut menjelaskan cara menanggapi permintaan ini.

Note

Ketika Anda membuat penawaran pribadi lebih tinggi dari \$250.000 melalui Portal Manajemen AWS Marketplace, persetujuan tambahan mungkin diperlukan. Untuk informasi lebih lanjut, hubungi Tim Sukses Penawaran Pribadi (POST), Vendor Finance Success (VFS), atau tim Pelaporan Penjual melalui [Portal Manajemen AWS Marketplace](#)

Untuk membuat penawaran pribadi untuk produk layanan profesional

1. Hubungi pelanggan untuk menyelesaikan pertanyaan yang Anda miliki tentang permintaan tersebut. Setujui ketentuan penawaran sebelum membuat penawaran pribadi di AWS Marketplace. Pembeli tidak berkewajiban untuk membeli produk Anda, jadi masuk akal untuk menyetujui sebelum membuat penawaran.
2. Buka browser web dan masuk ke file [Portal Manajemen AWS Marketplace](#).
3. Pilih Penawaran dari menu, lalu pilih Buat penawaran pribadi.
4. Pada halaman Buat penawaran pribadi, pilih produk yang ingin Anda buat penawaran pribadi. Hanya produk yang tersedia saat ini yang disertakan.
5. Masukkan ID akun Pembeli yang ingin Anda perpanjang penawaran pribadi. Anda dapat memasukkan hingga 25 akun dalam satu penawaran pribadi. Jika pembeli menggunakan fitur permintaan penawaran, pesan email yang Anda terima termasuk ID akun pembeli untuk akun yang meminta.
6. Pilih apakah akan mengizinkan pembeli membayar produk dengan mencicil. Biasanya, kontrak pendek dibayar dalam satu pembayaran. Kontrak yang lebih panjang mungkin memiliki opsi untuk membayar dengan mencicil, tetapi ini adalah bagian dari perjanjian yang Anda datangi dengan pembeli. Pilih Selanjutnya.
7. Lengkapi detail penawaran, termasuk informasi berikut:
 - Nama penawaran khusus — Berikan detail yang cukup sehingga Anda dan pelanggan akan mengenali penawaran tersebut. Sertakan nama perusahaan atau produk Anda dan deskripsi produk. Jangan menyertakan informasi identitas pribadi apa pun, termasuk nama, nomor telepon, atau alamat.
 - Tanggal akhir perjanjian — Tanggal akhir layanan yang disepakati berakhir. Misalnya, jika Anda menawarkan dukungan selama 1 tahun, masukkan tanggal yang berjarak 1 tahun dari tanggal layanan akan tersedia.
 - Dimensi produk — Harga dan unit untuk layanan yang Anda tawarkan, sebagai berikut:
 - Penawaran pembayaran sekaligus — Anda dapat mencantumkan masing-masing dimensi dengan harga yang terkait (misalnya, Anda dapat memiliki dimensi yang disebut Perak, Emas, dan Platinum). Pembeli dapat memilih dan membayar preferensi mereka.
 - Penawaran yang mencakup jadwal pembayaran — Anda harus memilih satu dimensi dan memberikan jadwal pembayaran dengan jumlah dan tanggal untuk setiap pembayaran.

 Note

Jika Anda ingin membuat penawaran nol dolar, Anda harus memilih Saya ingin mengaktifkan harga nol dolar untuk konfirmasi. Tindakan pencegahan ini membantu mencegah Anda membuat penawaran gratis secara tidak sengaja.

- Perjanjian layanan — Dokumen yang menentukan perjanjian layanan Anda dengan pelanggan. Dokumen yang Anda unggah (dalam format teks atau PDF) ditambahkan bersama ke dalam satu dokumen PDF, jadi pastikan bahwa nama file tidak diperlukan untuk memahami konten.
 - Tanggal kedaluwarsa penawaran — Tanggal penawaran berakhir. Ini menentukan berapa lama pembeli harus menerima penawaran dan tidak terkait dengan kapan layanan profesional akan tersedia. Anda dapat memperpanjang tanggal kedaluwarsa penawaran setelah penawaran Anda dibuat.
8. Pilih Berikutnya setelah Anda selesai mengedit opsi.
 9. Pada halaman Tinjau penawaran, pastikan detail penawaran sudah benar, lalu pilih Buat penawaran.

 Note

Penawaran Anda mungkin membutuhkan waktu untuk dipublikasikan. Setelah diterbitkan, Anda dapat melihat penawaran di halaman Kelola penawaran. Jika Anda perlu mengedit penawaran (yang belum diterima), Anda dapat melakukannya dari halaman itu.

10. Setelah penawaran diterbitkan, dan tersedia di halaman Kelola penawaran pribadi, dari menu Tindakan untuk penawaran itu, pilih Salin URL penawaran, lalu kirimkan dalam pesan email ke pembeli untuk menerima.

Edit informasi produk

Prosedur berikut menjelaskan cara mengedit informasi produk untuk produk layanan profesional yang ada di Portal Manajemen AWS Marketplace.

Untuk mengedit informasi produk

1. Buka browser web dan masuk ke file [Portal Manajemen AWS Marketplace](#).
2. Dari menu Produk, pilih Layanan profesional. Halaman ini menunjukkan kepada Anda semua produk layanan profesional yang telah Anda buat, serta permintaan apa pun yang Anda miliki untuk membuat atau memodifikasi produk ini.
3. Pilih produk yang sudah ada yang ingin Anda edit. Kemudian, dari menu Permintaan perubahan, pilih Perbarui informasi produk.
4. Buat perubahan pada detailnya. Untuk informasi selengkapnya tentang bidang yang dapat Anda edit, lihat [Memberikan rincian untuk produk layanan profesional di AWS Marketplace](#).
5. Pilih Kirim untuk membuat permintaan.
6. (Opsional) Jika Anda belum berada di tab Permintaan pada halaman Layanan profesional, lalu dari menu Produk Portal Manajemen AWS Marketplace, pilih Layanan profesional, lalu pilih tab Permintaan. Verifikasi bahwa Anda melihat permintaan Anda dengan judul Produk yang benar, dan bahwa status Permintaan sedang ditinjau. Produk Anda akan diperbarui dengan perubahan yang Anda minta dalam beberapa menit. Jika ada kesalahan, Anda dapat melihatnya di sini dan mengirimkan ulang hasil edit Anda setelah memperbaiki kesalahan.

Mengedit harga produk

Prosedur berikut menjelaskan cara mengedit informasi harga untuk produk layanan profesional yang ada di Portal Manajemen AWS Marketplace.

Untuk mengedit harga produk

1. Buka browser web dan masuk ke file [Portal Manajemen AWS Marketplace](#).
2. Dari menu Produk, pilih Layanan profesional. Halaman ini menunjukkan kepada Anda semua produk layanan profesional yang telah Anda buat, serta permintaan apa pun yang Anda buat untuk membuat atau memodifikasi produk ini.
3. Pilih produk yang sudah ada yang ingin Anda edit, lalu dari menu Permintaan perubahan, pilih Perbarui dimensi harga.

Note

Anda hanya dapat menambahkan dimensi harga baru melalui Portal Manajemen AWS Marketplace. Untuk mengubah atau menghapus dimensi yang dibuat sebelumnya,

hubungi [tim Operasi AWS Marketplace Penjual](#) dengan permintaan Anda. Dalam permintaan Anda, sertakan ID produk dan detail tentang dimensi apa yang ingin Anda ubah atau hapus.

4. Tambahkan dimensi harga baru yang Anda inginkan. Untuk informasi selengkapnya tentang bidang harga, lihat [Memberikan rincian untuk produk layanan profesional di AWS Marketplace](#).
5. Pilih Kirim untuk membuat permintaan.
6. (Opsional) Dari menu Produk Portal Manajemen AWS Marketplace, pilih Layanan profesional, lalu pilih tab Permintaan. Verifikasi bahwa Anda melihat permintaan Anda dengan judul Produk yang benar, dan bahwa status Permintaan sedang ditinjau. Produk Anda akan diperbarui dengan perubahan yang Anda minta dalam beberapa menit. Jika ada kesalahan, Anda dapat melihatnya di sini dan mengirimkan ulang hasil edit Anda setelah memperbaiki kesalahan.

Edit visibilitas produk

Secara default, produk dibuat dengan visibilitas terbatas — produk baru hanya terlihat dari akun Anda. Anda dapat menambahkan akun pengujian lainnya, atau membuat produk terlihat publik di AWS Marketplace. Prosedur berikut menjelaskan cara mengedit visibilitas produk layanan profesional yang ada di Portal Manajemen AWS Marketplace.

Untuk mengedit visibilitas produk

1. Buka browser web dan masuk ke file [Portal Manajemen AWS Marketplace](#).
2. Dari menu Produk, pilih Layanan profesional. Halaman ini menunjukkan kepada Anda semua produk layanan profesional yang telah Anda buat.
3. Pilih produk yang sudah ada yang ingin Anda edit atau perbarui visibilitas. Kemudian, dari menu Permintaan perubahan, pilih Perbarui visibilitas produk.
4. Pilih opsi untuk memperbarui visibilitas produk Anda dan pilih Kirim untuk mengirimkan permintaan Anda untuk ditinjau.
5. Verifikasi bahwa tab Permintaan menampilkan status Permintaan sebagai Dalam peninjauan. Ketika permintaan selesai, status menjadi Berhasil. Jika status Gagal, pilih nama permintaan untuk melihat Masalah Ditemukan.

Note

Untuk membuat produk terlihat di AWS Marketplace katalog publik memerlukan tinjauan produk oleh tim Operasi AWS Marketplace Penjual untuk memastikan bahwa produk memenuhi pedoman produk (lihat [Persyaratan untuk produk layanan profesional AWS Marketplace](#)). Permintaan dapat memakan waktu beberapa hari untuk diselesaikan.

Hapus produk layanan profesional

Prosedur berikut menjelaskan cara menghapus produk layanan profesional yang ada dari Portal Manajemen AWS Marketplace.

Note

Menghapus produk layanan profesional tidak akan memengaruhi penawaran pribadi aktif.

Untuk menghapus produk

1. Buka browser web dan masuk ke file [Portal Manajemen AWS Marketplace](#).
2. Dari menu Produk, pilih Layanan profesional. Halaman ini menunjukkan kepada Anda semua produk layanan profesional yang telah Anda buat.
3. Pilih produk yang sudah ada yang ingin Anda hapus. Kemudian, dari menu Permintaan perubahan, pilih Perbarui visibilitas produk.
4. Pilih Dibatasi sebagai opsi visibilitas untuk memperbarui visibilitas produk Anda dan pilih Kirim.
5. Verifikasi bahwa tab Permintaan menampilkan status Permintaan sebagai Dalam peninjauan. Ketika permintaan selesai, status menjadi Berhasil.

Note

Permintaan dapat memakan waktu beberapa hari untuk diselesaikan. Produk dengan penawaran aktif akan dipindahkan ke status terbatas hingga langganan atau kontrak aktif terakhir selesai dan kemudian dihapus AWS Marketplace. Status terbatas berarti bahwa pengguna yang ada dapat terus menggunakan produk. Namun, produk tidak akan lagi terlihat oleh publik atau tersedia untuk pengguna baru.

Memberikan rincian untuk produk layanan profesional di AWS Marketplace

Ketika Anda mempublikasikan produk layanan profesional AWS Marketplace, Anda harus menyediakan metadata produk. Metadata produk mencakup pengaturan produk untuk visibilitas dan harga serta informasi produk yang memberi pembeli informasi lebih lanjut tentang produk layanan profesional Anda. Anda dapat mengedit deskripsi produk, sumber daya tambahan, dimensi harga, dan visibilitas produk. Bagian berikut memberikan informasi yang dapat Anda gunakan saat menyiapkan atau mengedit detail produk Anda.

Note

Untuk informasi tentang pedoman dan persyaratan untuk produk layanan profesional, lihat [Persyaratan untuk produk layanan profesional AWS Marketplace](#).

Topik

- [Deskripsi produk](#)
- [Sumber daya tambahan](#)
- [Informasi Dukungan](#)
- [Dimensi harga](#)
- [Visibilitas produk](#)

Deskripsi produk

Bagian deskripsi produk dalam detail produk adalah inti dari produk Anda. Ini menggambarkan produk Anda kepada pembeli potensial Anda sehingga mereka dapat membuat keputusan pembelian. Bagian detail produk ini mencakup data berikut:

- Judul produk — Nama produk Anda. Ini digunakan untuk mengidentifikasi produk Anda; itu terlihat di halaman produk dan dalam hasil pencarian. Berikan nama yang berarti untuk produk Anda. Itu harus unik di dalam AWS Marketplace.
- SKU - (Opsional) Digunakan untuk melacak produk Anda. AWS Marketplace Informasi ini untuk Anda gunakan sendiri; pembeli tidak melihatnya.

- Deskripsi singkat — Deskripsi singkat tentang produk Anda yang muncul di ubin dan di bawah judul produk dalam katalog produk. AWS Marketplace
- Deskripsi panjang — Deskripsi yang lebih panjang dan diformat yang menjelaskan detail produk Anda kepada pembeli. Buat daftar fitur produk, manfaat, penggunaan, dan informasi lain yang spesifik untuk produk. Gunakan format yang tersedia untuk membuat informasi lebih mudah dipahami dan dipindai.
- Logo produk - Bidang ini adalah URL Amazon S3 publik yang menunjuk ke file gambar yang mewakili produk Anda. File harus dalam format.png, .jpg, atau .gif, dengan latar belakang transparan atau putih, di bawah 5MB, dan lebar antara 110-10.000 piksel dan tinggi. Logo diunggah selama pengiriman produk, dan disimpan di. AWS Marketplace Memodifikasi isi URL tidak akan mengubah logo AWS Marketplace setelah dikirimkan.

Note

URL Amazon S3 yang Anda berikan harus tersedia untuk umum. Ini adalah properti bucket Amazon S3 tempat file berada. Untuk informasi selengkapnya, lihat [Bagaimana cara mengedit setelan akses publik untuk bucket Amazon S3?](#) di Panduan Pengguna Konsol Layanan Penyimpanan Sederhana Amazon.

- Sorotan — Satu set satu hingga tiga poin pendek tentang produk Anda, yang menjelaskan fitur atau pembeda utamanya. Setidaknya satu sorotan diperlukan.
- Kategori produk — Jenis layanan yang Anda berikan. Anda harus memilih setidaknya satu, dan hingga tiga, kategori. Ada banyak kategori untuk dipilih, tetapi produk layanan profesional harus mencakup setidaknya satu dari yang berikut:

Penilaian

Evaluasi lingkungan operasi pelanggan saat ini untuk menemukan solusi yang tepat untuk organisasi mereka.

Implementasi

Bantuan dengan konfigurasi, penyiapan, dan deployment perangkat lunak pihak ketiga.

Dukungan premium

Akses ke bimbingan dan bantuan dari para ahli, yang dirancang untuk kebutuhan pelanggan.

Layanan terkelola

End-to-end manajemen lingkungan atas nama pelanggan.

Pelatihan

Lokakarya, program, dan alat pendidikan yang disesuaikan yang disediakan oleh para ahli untuk membantu karyawan pelanggan mempelajari praktik terbaik.

- Kata kunci untuk hasil pencarian — Berikan hingga tiga kata kunci yang mungkin digunakan pembeli untuk mencari produk Anda. Anda dapat mencantumkan kata kunci dalam daftar yang dipisahkan koma, hingga 250 karakter.
- Produk terkait — opsional — Sertakan setidaknya satu produk publik dari AWS Marketplace mana layanan Anda bekerja dengan atau mendukung. AWS Marketplace menggunakan produk ini sebagai masukan saat memilih produk untuk ditampilkan di halaman detail produk Anda atau di Produk terkait untuk produk tersebut.

Sumber daya tambahan

Di bagian Sumber daya tambahan pada detail produk, Anda dapat memberikan tautan ke sumber daya yang telah Anda buat untuk membantu pelanggan Anda. Ini adalah satu set opsional dari satu hingga tiga sumber daya yang dapat diunduh yang disimpan secara online. Contoh sumber daya termasuk lembar informasi produk, whitepaper, atau manual produk. Untuk setiap sumber daya, berikan nama dan URL untuk sumber daya.

Informasi Dukungan

Bagian ini adalah bidang teks berformat yang memungkinkan Anda menjelaskan dukungan yang Anda berikan untuk layanan Anda.

Pelanggan mengharapkan dukungan pada masalah, seperti menggunakan layanan, pemecahan masalah, dan meminta pengembalian uang (jika ada). Anda harus memberikan setidaknya satu dari berikut ini untuk dukungan produk Anda: alamat email, nomor telepon, atau URL dukungan. Sertakan deskripsi tingkat dukungan yang dapat diharapkan pembeli ketika mereka membeli produk Anda.

Dimensi harga

Dimensi harga untuk layanan profesional adalah paket yang Anda tawarkan. Misalnya Anda mungkin menawarkan dukungan Silver, Gold, dan Platinum. Atau Anda mungkin menawarkan 10, 20, atau 50 jam konsultasi. Untuk setiap dimensi yang ingin Anda tawarkan (setidaknya satu, hingga 24), tentukan nama dan deskripsi. Ketika Anda membuat penawaran pribadi untuk produk dengan bekerja dengan pembeli secara langsung, Anda menetapkan harga aktual untuk dimensi ini.

Note

Untuk informasi tentang bagaimana dimensi harga digunakan, dan bagaimana harga ditetapkan, lihat [Buat penawaran pribadi](#).

Visibilitas produk

Produk yang dirilis dapat dilihat hanya di AWS Marketplace akun Anda sendiri, ke satu set kecil akun uji, atau ke semua AWS akun. Secara default, produk diterbitkan dalam rilis pribadi. Untuk mengubah visibilitas produk, lihat [Edit visibilitas produk](#).

Persyaratan untuk produk layanan profesional AWS Marketplace

AWS Marketplace mempertahankan persyaratan untuk semua produk dan penawaran di AWS Marketplace. Persyaratan ini membantu mempromosikan katalog digital kurasi yang aman, terjamin, dan dapat dipercaya untuk pelanggan kami. Kami mendorong penjual untuk meninjau implementasi kontrol dan protokol tambahan yang berlaku untuk memenuhi kebutuhan produk spesifik mereka.

AWS Marketplace meninjau semua produk dan metadata terkait ketika dikirimkan untuk memastikan bahwa mereka memenuhi atau melampaui persyaratan saat ini AWS Marketplace. Kami meninjau dan menyesuaikan persyaratan ini untuk memenuhi persyaratan keamanan kami. Selain itu, AWS Marketplace terus meninjau produk untuk memverifikasi bahwa mereka memenuhi setiap perubahan pada persyaratan ini. Jika produk tidak sesuai, AWS Marketplace hubungi Anda untuk memperbarui produk Anda. Dalam beberapa kasus, produk Anda mungkin sementara tidak tersedia untuk pelanggan baru sampai masalah diselesaikan.

Topik

- [Pedoman pengaturan produk](#)
- [Persyaratan informasi pelanggan](#)
- [Pedoman penggunaan produk](#)
- [Pedoman arsitektur](#)

Pedoman pengaturan produk

Semua produk layanan profesional harus mematuhi pedoman pengaturan produk berikut:

- Semua dimensi harga harus berhubungan dengan layanan aktual yang ditawarkan dan ditagih menggunakan AWS Marketplace.
- Produk Anda harus terdaftar dalam salah satu kategori ini: Penilaian, Implementasi, Layanan terkelola, dukungan Premium, atau Pelatihan.
- Selain kategori layanan profesional yang diperlukan, produk Anda juga harus dikategorikan dengan benar dengan memilih kategori lain yang sesuai dengan layanan yang ditawarkan.
- Logo produk Anda tidak boleh dirancang untuk membingungkan dengan AWS logo, atau logo apa pun dari pihak ketiga yang tidak terkait.
- Detail produk Anda tidak boleh mengandung materi yang menyinggung atau eksplisit. Mereka harus mematuhi Kebijakan Penggunaan yang AWS Dapat Diterima yang tersedia di <https://aws.amazon.com/aup/>.
- Produk layanan profesional Anda harus secara langsung mendukung atau menawarkan layanan yang terkait dengan produk perangkat lunak pihak ketiga yang tercantum pada AWS Marketplace atau membantu pelanggan mencapai hasil spesifik yang terkait dengan adopsi atau pengelolaan mereka AWS Cloud.

Persyaratan informasi pelanggan

Semua produk layanan profesional harus mematuhi persyaratan informasi pelanggan berikut:

- Produk layanan profesional harus ditagih seluruhnya melalui dimensi yang tercantum pada AWS Marketplace.
- Anda tidak diizinkan untuk mengumpulkan informasi pembayaran pelanggan untuk daftar produk layanan profesional Anda AWS Marketplace kapan saja, termasuk informasi kartu kredit dan rekening bank.
- Setiap informasi pelanggan atau calon pelanggan yang diberikan AWS kepada Anda sehubungan dengan produk layanan profesional Anda harus digunakan semata-mata sesuai dengan Syarat dan Ketentuan untuk Penjual. AWS Marketplace

Pedoman penggunaan produk

Semua produk layanan profesional harus mematuhi pedoman penggunaan produk berikut:

- Setelah pelanggan menghubungi Anda melalui daftar layanan profesional, Anda harus menghubungi mereka dalam waktu dua hari kerja. Setelah pelanggan menerima penawaran

pribadi, Anda harus menghubungi mereka atau memberi mereka langkah selanjutnya dalam waktu dua hari kecuali dinyatakan lain dalam penawaran pribadi.

- Pelanggan harus dapat dengan mudah mendapatkan bantuan dengan masalah, seperti menggunakan layanan, pemecahan masalah, dan meminta pengembalian uang (jika ada). Opsi kontak Dukungan harus ditentukan pada halaman arahan pemenuhan. Deskripsi dukungan harus berisi pernyataan tentang tingkat dukungan yang dapat diharapkan pelanggan.
- Gambaran umum produk Anda harus mencakup proposisi nilai yang jelas, fitur utama, tautan ke dokumentasi terperinci, dan definisi yang jelas tentang dukungan pra-pembelian dan pasca-pembelian dari layanan yang ditawarkan.
- Produk Anda harus memiliki persyaratan layanan yang jelas dan mudah.

Pedoman arsitektur

Semua produk layanan profesional harus mematuhi pedoman arsitektur berikut:

- Produk layanan profesional harus terkait dengan AWS layanan atau setidaknya satu AWS Marketplace produk publik (hingga empat) di mana produk menawarkan layanan untuk produk terkait secara langsung atau menawarkan layanan yang mendorong lebih banyak pelanggan ke produk terkait tersebut.
- Jika layanan yang ditawarkan memerlukan sumber daya tambahan dalam infrastruktur pelanggan, ikuti panduan ini:
 - Menyediakan sumber daya dengan cara yang aman, seperti dengan menggunakan AWS Security Token Service atau AWS Identity and Access Management (IAM).
 - Berikan dokumentasi tambahan termasuk deskripsi semua AWS layanan yang disediakan, pernyataan kebijakan IAM, dan bagaimana peran IAM atau pengguna digunakan dan digunakan di akun pelanggan.
 - Sertakan pemberitahuan dalam deskripsi produk yang menjelaskan bahwa jika pelanggan mengeluarkan biaya AWS infrastruktur tambahan, terpisah dari AWS Marketplace transaksi mereka, mereka bertanggung jawab untuk membayar biaya infrastruktur tambahan.
 - Jika produk Anda menyebarkan agen, berikan instruksi kepada pelanggan yang menjelaskan cara menyebarkan agen di dalamnya. Akun AWS

Harga produk layanan profesional di AWS Marketplace

Sebagai AWS Marketplace penjual, Anda dapat menerima penawaran pribadi untuk produk layanan profesional. Penawaran pribadi adalah istilah yang dinegosiasikan yang digunakan untuk membeli produk dari AWS Marketplace. Sebagai penjual, Anda bernegosiasi dengan pembeli untuk sampai pada penawaran pribadi. Penawaran pribadi adalah salah satu model harga yang tersedia di AWS Marketplace. Dengan penawaran pribadi penjual, ada opsi yang tersedia untuk kontrak multi-tahun dan durasi khusus. Topik ini memberikan informasi lebih lanjut tentang harga produk layanan profesional dan penawaran pribadi.

Untuk informasi selengkapnya tentang kontrak multi-tahun dan durasi khusus, lihat [Mempersiapkan penawaran pribadi untuk AWS Marketplace produk Anda](#) dan [Paket cicilan penawaran pribadi](#).

Anda hanya dapat menetapkan satu harga per produk. Untuk informasi lebih lanjut tentang harga AWS Marketplace produk, lihat [Harga produk untuk AWS Marketplace](#).

Bagaimana penawaran pribadi bekerja

Anda dapat membuat dan mengelola penawaran pribadi Anda dari halaman Penawaran di [Portal Manajemen AWS Marketplace](#). Anda menentukan produk untuk penawaran untuk menghasilkan ID dan URL unik. Anda akan membuat rencana harga untuk penawaran pribadi, menambahkan persyaratan hukum dan dokumen penjualan, dan memperluas penawaran ke pembeli tertentu Akun AWS. Penawaran ini hanya terlihat oleh akun tempat Anda membuat penawaran.

Setelah Anda membuat penawaran pribadi dan memberi tahu pembeli potensial, mereka dapat melihat dan menerima penawaran tersebut. Untuk melihat penawaran, pembeli harus masuk ke Akun AWS yang menerima penawaran.

Note

Pembeli tidak dapat melihat penawaran kecuali Anda memperluasnya ke akun tertaut atau akun manajemen mereka. Anda tidak dapat memberikan batasan layanan dalam penawaran, sehingga pembeli dapat menggunakan sebanyak mungkin produk Anda dengan harga yang dinegosiasikan seperti yang mereka inginkan, kecuali produk memiliki batas.

Untuk informasi tentang membuat penawaran pribadi, lihat [Membuat dan mengelola penawaran pribadi](#).

Penawaran pribadi dilacak dalam laporan penjual. Untuk informasi selengkapnya, lihat [Melaporkan penawaran pribadi](#) dan [panduan laporan Penjual](#).

Produk data

AWS Data Exchange adalah layanan yang memudahkan AWS pelanggan untuk bertukar set data berbasis file dengan aman di file. AWS Cloud Sebagai penyedia, AWS Data Exchange menghilangkan kebutuhan untuk membangun dan memelihara teknologi pengiriman, hak, atau penagihan data apa pun. Penyedia di AWS Data Exchange memiliki saluran yang aman, transparan, dan andal untuk menjangkau AWS pelanggan dan memberikan langganan kepada pelanggan yang sudah ada dengan lebih efisien. Proses untuk menjadi penyedia AWS Data Exchange memerlukan beberapa langkah untuk menentukan kelayakan.

Produk data memiliki bagian-bagian berikut:

- Detail produk — Informasi ini membantu calon pelanggan memahami apa produk itu. Ini termasuk nama, deskripsi (pendek dan panjang), gambar logo, dan informasi kontak dukungan. Detail produk diisi oleh penyedia.
- Penawaran produk — Untuk membuat produk tersedia di AWS Data Exchange, penyedia harus menentukan penawaran umum. Ini termasuk harga dan durasi, perjanjian berlangganan data, kebijakan pengembalian dana, dan opsi untuk membuat penawaran khusus.
- Kumpulan data — Sebuah produk dapat berisi satu atau lebih kumpulan data. Kumpulan data adalah kumpulan dinamis konten data berbasis file. Kumpulan data bersifat dinamis dan berversi menggunakan revisi. Setiap revisi dapat berisi beberapa aset.

Untuk informasi selengkapnya, termasuk persyaratan kelayakan, lihat [Menyediakan produk data di AWS Data Exchange](#) di Panduan Pengguna AWS Data Exchange.

Mengirimkan produk Anda untuk dipublikasikan AWS Marketplace

Anda menggunakan proses pengiriman produk untuk membuat produk Anda tersedia di AWS Marketplace. Produk bisa sederhana, seperti Amazon Machine Image (AMI) tunggal dengan satu struktur harga, atau lebih rumit, dengan AWS CloudFormation template, dan opsi harga yang kompleks dan jadwal pembayaran. Anda menentukan penawaran produk Anda dan mengirimkannya melalui salah satu dari dua cara: Portal Manajemen AWS Marketplace

- Menggunakan tab Produk — Untuk produk yang kurang kompleks, Anda menggunakan tab Produk untuk sepenuhnya menentukan dan mengirimkan permintaan Anda.
- Menggunakan tab Aset — Untuk produk yang lebih kompleks dan memerlukan definisi lebih lanjut, Anda mengunduh formulir pemuatan produk (PLF), menambahkan detail produk, dan kemudian mengunggah formulir yang sudah diisi menggunakan opsi Unggah file.

Note

Penyedia produk data harus menggunakan konsol AWS Data Exchange untuk mempublikasikan produk. Untuk informasi selengkapnya, lihat [Menerbitkan produk baru](#) di Panduan Pengguna AWS Data Exchange.

Kami menyarankan Anda memulai dengan menggunakan tab Produk untuk menentukan pendekatan mana yang akan digunakan. Tabel berikut mencantumkan konfigurasi dan pendekatan yang Anda gunakan untuk mengirimkan permintaan Anda. Kolom pertama adalah model harga untuk produk Anda, dan tiga kolom lainnya menjelaskan bagaimana produk dikerahkan ke pelanggan.

Model penentuan harga	Produk diluncurkan menggunakan AMI simpul tunggal	Produk diluncurkan dengan AWS CloudFormation	Produk diluncurkan sebagai perangkat lunak sebagai layanan (SaaS)
Bring Your Own License (BYOL)	Tab produk	Tab Aset	

Model penentuan harga	Produk diluncurkan menggunakan AMI simpul tunggal	Produk diluncurkan dengan AWS CloudFormation	Produk diluncurkan sebagai perangkat lunak sebagai layanan (SaaS)
Kosong	Tab produk	Tab Aset	
Dibayar Per Jam	Tab produk	Tab Aset	
Dibayar Setiap Jam dengan Tahunan	Tab produk	Tab Aset	
Dibayar Bulanan	Tab produk	Tab Aset	
Setiap Jam dengan Bulanan	Tab Aset	Tab Aset	
Penggunaan Berbayar (AWS Marketplace Metering Service)	Tab produk	Tab Aset	
Harga Kontrak	Tab produk		
Berlangganan SaaS			Tab produk
Kontrak SaaS			Tab produk
Warisan SaaS			Tab Aset

Anda dapat mengirimkan produk satu per satu atau, jika Anda menggunakan formulir pemuatan produk, Anda dapat mengirimkan beberapa produk atau pembaruan produk secara bersamaan. Anda tidak dapat menggunakan tab Produk untuk mengirimkan beberapa produk. Jika Anda tidak jelas tentang produk mana yang harus dikirimkan dan cara mengirimkannya, mulailah dengan menggunakan tab Produk. Jika Anda memiliki masalah dalam membuat kiriman, hubungi tim [Operasi AWS Marketplace Penjual](#).

Topik

- [Menggunakan tab Produk](#)

- [Persyaratan logo perusahaan dan produk](#)
- [Persyaratan untuk mengirimkan perangkat lunak paket ulang berbayar](#)
- [Persyaratan untuk produk dengan komponen perangkat keras](#)
- [AWS CloudFormation produk yang diluncurkan \(gratis atau berbayar\) atau produk AMI berbayar berbasis penggunaan](#)
- [Perubahan dan pembaruan produk](#)
- [Waktu dan harapan](#)
- [AMIs Mengirimkan ke AWS Marketplace](#)
- [Daftar periksa akhir](#)

Menggunakan tab Produk

Untuk mengakses tab Produk, masuk ke file Portal Manajemen AWS Marketplace. Dari tab Produk, pilih Server, SaaS, atau Pembelajaran mesin, tergantung pada jenis produk yang Anda kelola.

Dasbor untuk jenis produk tersebut muncul dan menampilkan produk Anda saat ini. Jika Anda memilih tab Permintaan, dasbor menampilkan permintaan yang belum selesai dan riwayat permintaan Anda yang telah selesai. Setelah Anda mulai membuat permintaan produk, Anda dapat menyimpan pekerjaan Anda dalam proses, dan jika perlu, buat permintaan Anda dalam beberapa sesi berbeda.

Saat Anda mengirimkan permintaan produk Anda, AWS Marketplace tim meninjaunya. Anda dapat memantau status permintaan Anda di halaman produk untuk jenis produk yang Anda minta. Untuk produk baru, setelah permintaan Anda disetujui untuk dipublikasikan, Anda menerima URL daftar terbatas yang Anda gunakan untuk melihat pratinjau dan menyetujui kiriman Anda. Penawaran produk Anda tidak dipublikasikan sampai Anda menyetujui pengajuan.

Ketika Anda meminta pembaruan produk, itu diterbitkan tanpa perlu bagi Anda untuk meninjau dan menyetujui perubahan. Ini termasuk menambahkan atau menghapus versi, dan perubahan metadata.

Anda melacak status permintaan Anda di bawah tab Permintaan. Tab menampilkan salah satu dari berikut ini:

- Draf — Anda telah memulai proses permintaan tetapi belum mengirimkan permintaan Anda.
- Dikirim - Anda telah menyelesaikan dan mengirimkan permintaan Anda, dan itu sedang ditinjau.
- Tindakan yang Diperlukan - AWS Marketplace Tim telah meninjau permintaan Anda dan membutuhkan informasi lebih lanjut.

- **Persetujuan Diperlukan** — AWS Marketplace Tim telah membuat URL daftar terbatas untuk produk Anda. Anda harus meninjau dan menyetujui atau menolak URL sebelum AWS Marketplace akan mempublikasikan. Jika Anda menyetujui, status berubah menjadi Publishing Pending saat situs dipublikasikan. Jika Anda menolak, status kembali ke Draf sehingga Anda dapat mengubah permintaan.
- **Penerbitan Tertunda** — Anda telah menyetujui mock-up permintaan Anda dan AWS Marketplace menerbitkan produk Anda.
- **Kedaluwarsa** — Anda memulai proses permintaan tetapi tidak menyelesaikannya dalam waktu enam bulan, sehingga permintaan tersebut kedaluwarsa.

Jika Anda memiliki entri dengan status Dikirim, Anda dapat menarik kembali kiriman. Jika Anda memiliki entri dengan status Draf, Anda dapat menghapus permintaan tersebut. Ini akan memungkinkan Anda untuk memulai dari awal. Saat Anda menghapus entri Draf, entri dipindahkan ke tab Riwayat Permintaan.

Untuk menambahkan produk Anda di AWS GovCloud (US) Wilayah AWS, Anda harus [memiliki akun aktif AWS GovCloud \(AS\)](#) dan mematuhi AWS GovCloud (US) persyaratan, termasuk persyaratan kontrol ekspor.

Persyaratan logo perusahaan dan produk

Logo perusahaan Anda dan logo untuk produk Anda harus sesuai dengan AWS Marketplace pedoman berikut sehingga pengalaman pengguna seragam saat menjelajah AWS Marketplace:

Spesifikasi logo produk — Gambar logo produk Anda harus memiliki latar belakang transparan atau putih dan berukuran 120 hingga 640 piksel, dengan rasio 1:1 atau 2:1 (lebar).

Spesifikasi logo perusahaan — Gambar logo perusahaan Anda harus memiliki latar belakang transparan dan berukuran 220 x 220 piksel, memungkinkan bantalan 10 piksel di setiap sisi di dalamnya.

Persyaratan untuk mengirimkan perangkat lunak paket ulang berbayar

Sebelum Anda dapat mengirimkan daftar untuk perangkat lunak yang dikemas ulang, Anda harus memenuhi persyaratan berikut. Dalam hal ini, perangkat lunak yang dikemas ulang mencakup

sumber terbuka AMIs atau perangkat lunak yang dibuat oleh vendor lain, seperti AMI dengan Windows.

Persyaratan

- Judul produk harus menunjukkan nilai tambah dengan pengemasan ulang Anda. <Product2>Contoh judul produk meliputi: Hardened<Product>, dengan paket tambahan<Product>, atau pada<Product1>.
- Judul produk tidak boleh mengandung bahasa lain yang tidak didukung dengan dokumentasi. Misalnya, judul produk tidak boleh menggunakan kata-kata bersertifikat, asli, atau gratis kecuali jika ini dibuktikan dalam detail produk yang Anda berikan.
- Deskripsi singkat produk harus menyertakan pernyataan yang jelas yang merangkum biaya produk. Deskripsi singkat harus dimulai dengan frasa Produk ini memiliki biaya yang terkait dengannya untuk... . Misalnya, jika suatu produk menyertakan biaya untuk dukungan dari penjual, maka deskripsi produk harus menyatakan: Produk ini memiliki biaya yang terkait dengannya untuk dukungan penjual.
- Logo produk harus sama dengan logo perusahaan yang digunakan selama proses pendaftaran penjual Anda. Logo produk dapat berbeda dari logo perusahaan Anda hanya jika Anda menggunakan logo perangkat lunak resmi, di mana Anda harus menerima izin eksplisit dari vendor perangkat lunak asli. Jika izin eksplisit diperoleh, tautan ke dokumentasi tersebut harus disertakan di bagian catatan permintaan perubahan, atau di kolom Masukkan deskripsi singkat pada halaman Unggahan File saat menggunakan formulir pemuatan produk.
- Untuk produk AMI, nama AMI tidak boleh digunakan kembali dari produk aslinya. Nama AMI harus dimulai dengan nama penjual dan ikuti format ini: [Nama Penjual] [name-given-to-ami].

Jika nama AMI tidak mematuhi konvensi penamaan, Anda dapat menyalin AMI dari AWS konsol dan mengganti namanya. Untuk informasi selengkapnya, lihat [Menyalin EC2 AMI Amazon](#), di Panduan EC2 Pengguna Amazon.

Jika daftar berbayar adalah untuk produk perangkat lunak mandiri yang tidak dibuat oleh perusahaan Anda dan tidak ada kekayaan intelektual yang ditambahkan ke produk, seperti menggabungkan pustaka perangkat lunak tambahan atau menambahkan konfigurasi khusus, maka bersama dengan persyaratan sebelumnya, Anda juga harus memenuhi persyaratan berikut:

- Judul produk harus menyertakan nama penjual (bersama dengan nilai tambah, seperti yang dijelaskan sebelumnya). Nama penjual adalah nama yang digunakan selama pendaftaran penjual. Misalnya, dengan dukungan pemeliharaan oleh <Product><seller>.

- Baris pertama dari deskripsi panjang produk harus dimulai dengan frasa Ini adalah produk perangkat lunak yang dikemas ulang di mana biaya tambahan berlaku untuk... (atau, jika itu open source, Ini adalah produk perangkat lunak open source yang dikemas ulang di mana biaya tambahan berlaku untuk...). Kemudian, deskripsi panjang harus menyertakan pernyataan yang jelas yang merangkum apa yang Anda kenakan, serta detail tambahan yang menjelaskan fitur-fitur tersebut. Misalnya, deskripsi panjang tentang pengisian produk open source untuk dukungan tambahan mungkin dimulai sebagai: Ini adalah produk perangkat lunak open source yang dikemas ulang di mana biaya tambahan berlaku untuk dukungan dengan {SLA Details}.

Persyaratan untuk produk dengan komponen perangkat keras

Penjualan produk perangkat keras tidak diizinkan AWS Marketplace. Jika Anda mengirimkan produk perangkat lunak yang memerlukan komponen perangkat keras (misalnya, kartu SIM, perangkat pintar, perangkat IoT, atau sensor), Anda harus memenuhi persyaratan berikut:

- Komponen perangkat keras tidak dapat dijual AWS Marketplace.
- Biaya komponen perangkat keras tidak dapat dimasukkan dalam harga daftar produk Anda.
- Bagian Ikhtisar Produk dari daftar harus menyertakan pernyataan berikut: Perangkat keras apa pun yang mungkin diperlukan dengan daftar ini harus diperoleh secara terpisah. Tinjau detail produk untuk informasi lebih lanjut.

AWS CloudFormation produk yang diluncurkan (gratis atau berbayar) atau produk AMI berbayar berbasis penggunaan

Note

Beberapa model harga tidak lagi mengharuskan Anda menggunakan formulir pemuatan produk yang dijelaskan di bagian ini untuk mempublikasikan AMI dengan CloudFormation produk. Saat Anda membuat Amazon Machine Image (AMI) atau AMI dengan CloudFormation halaman [produk server](#) di portal penjual, dan tidak segera diminta untuk mengunduh formulir pemuatan produk, lihat [Membuat produk berbasis AMI](#) dan [Menambahkan CloudFormation templat ke](#) daftar Anda.

Gunakan formulir beban produk (PLF) untuk mengirimkan produk yang diluncurkan AWS Marketplace pelanggan dengan menggunakan AWS CloudFormation templat. PLF tersedia melalui Portal Manajemen AWS Marketplace

Anda mengikuti langkah-langkah luas ini untuk mengirimkan produk:

- Pilih model harga.
- Unduh formulir pemuatan produk (PLF), spreadsheet Microsoft Excel.
- Isi formulir pemuatan produk.
- Ikuti instruksi dalam formulir untuk mengirimkan produk Anda.

Untuk informasi lebih lanjut tentang menyelesaikan setiap langkah, perluas bagian dalam urutan yang tercantum.

Memilih model penetapan harga

Anda harus memilih model harga untuk produk Anda. Model yang Anda pilih mengontrol informasi harga yang Anda masukkan ke dalam PLF. Untuk daftar model harga yang didukung, lihat [Harga produk AMI untuk AWS Marketplace](#) di panduan ini.

Mengunduh PLF untuk produk baru

1. Mulai [Portal AWS Marketplace Manajemen](#).
2. Pada tab Aset, di panel sebelah kanan, pilih [AMI Tunggal dengan tautan CloudFormation produk](#).

Formulir muncul di browser Anda.

3. Pilih Unduh file, lalu edit file di Excel.

— atau —

Jika Anda memiliki Microsoft OneDrive, pilih Edit salinan. Itu menyimpan PLF ke OneDrive, dan Anda dapat mengeditnya di sana.

Note

Spreadsheet berisi beberapa contoh produk. Anda harus menghapusnya sebelum mengirimkan formulir.

Mengunduh PLF untuk produk yang sudah ada

Untuk mengunduh formulir

- Mulai Dasbor AWS Marketplace dan pilih Unduh formulir pemuatan produk.



Spreadsheet berisi data untuk versi produk Anda sebelumnya. Biarkan data itu di tempatnya dan tambahkan produk baru pada baris kosong berikutnya.

Mengisi PLF

Langkah-langkah berikut menjelaskan cara melengkapi formulir beban produk (PLF). Langkah-langkah tersebut berlaku untuk produk baru dan yang sudah ada.

Untuk mengisi formulir

1. Di kolom SKU to Refund and Cancellation Policy, masukkan semua informasi yang relevan dengan produk Anda.

Note

Di kolom Petunjuk Akses Produk, Anda harus memberikan petunjuk penggunaan yang terperinci dan jelas. Ikuti persyaratan yang tercantum di [Membuat AMI dan petunjuk penggunaan produk kontainer untuk AWS Marketplace](#) dalam panduan ini.

2. Di kolom URL Relatif Type to Endpoint URL, masukkan informasi yang diperlukan untuk AMI Anda.

⚠ Important

Anda harus berbagi AMI Anda dengan AWS Marketplace. Untuk melakukan itu, ikuti langkah-langkah di [AMIs Mengirimkan ke AWS Marketplace](#) dalam panduan ini.

3. Kolom URL Perjanjian Lisensi Pengguna Akhir menyediakan tautan ke perjanjian AWS Marketplace standar. Anda dapat menerima perjanjian itu atau memasukkan tautan ke EULA yang Anda sukai untuk digunakan. Jika Anda memberikan tautan, itu harus memungkinkan pelanggan untuk mengunduh EULA, seperti tautan dari ember Amazon S3.

Untuk informasi lebih lanjut tentang kontrak standar, lihat [Menggunakan kontrak standar di AWS Marketplace](#) di panduan ini.

4. Di kolom us-east-1 Availability to Make available for all future instance types, **TRUE** masukkan **FALSE** atau di bawah masing-masing Wilayah yang ingin Anda AWS gunakan.

ℹ Note

GovCloud Daerah memiliki persyaratan tambahan. Misalnya, Anda harus memiliki GovCloud akun untuk menggunakan GovCloud Wilayah. Untuk informasi selengkapnya, lihat [Menyiapkan](#), di Panduan AWS GovCloud Pengguna.

5. Di kolom Jenis Instance yang Direkomendasikan, terima jenis instans yang direkomendasikan atau pilih yang lain dari daftar. Pastikan jenis instance tersedia di Wilayah yang ingin Anda gunakan.

ℹ Note

- Sebagian besar pelanggan menerima jenis instans yang direkomendasikan.
- Anda harus memastikan bahwa jenis instans tersedia di Wilayah yang sama dengan produk.

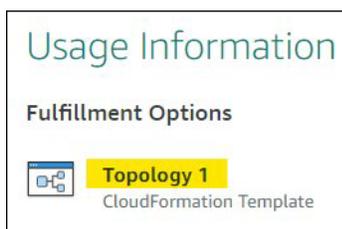
6. Di kolom antara Jenis Instance yang Direkomendasikan dan Negara yang Akan Disertakan, masukkan **TRUE** di bawah jenis instans Anda. Ini mengaktifkan jenis instance. Masukkan **FALSE** untuk jenis yang tersisa. Untuk informasi selengkapnya tentang tipe instance, lihat <https://aws.amazon.com/ec2/instance-types/>.

7. Di Negara yang akan disertakan dan Negara untuk mengecualikan kolom, masukkan kode negara dua huruf, seperti AS, dari negara mana pun yang ingin Anda sertakan atau kecualikan.
8. Di kolom Model Harga, masukkan model harga untuk produk Anda.

Daftar berikut menjelaskan model harga dan kolom tambahan yang harus Anda selesaikan.

- URL BYOL - Masukkan URL lisensi. Tidak perlu memasukkan informasi harga.
 - Per Jam — Masukkan harga untuk jenis instans apa pun yang Anda tetapkan ke TRUE. Biarkan semua kolom lainnya kosong. Kolom yang terkait dengan ini adalah dari Harga Per Jam 1.medium hingga z1d.metal Harga Per Jam
 - Harga Tahunan Per Jam - Isi kolom yang tercantum pada langkah sebelumnya, ditambah kolom yang dimulai dari Harga Tahunan a1.medium hingga z1d.metal Harga Tahunan. Masukkan harga untuk jenis instans apa pun yang Anda tetapkan ke TRUE. Anda dapat membiarkan semua kolom lainnya kosong.
 - Penggunaan - Masukkan informasi yang terkait dengan dimensi penggunaan di kolom Kategori FCP ke Tingkat Dimensi 24 FCP.
 - Kontrak — Dalam kolom Kategori Kontrak hingga Dimensi Kontrak 24 Tarif 36 Bulan, masukkan informasi yang terkait dengan dimensi kontrak.
9. Di kolom Aturan Grup Keamanan 1 hingga Aturan Grup Keamanan 12, masukkan informasi tentang grup keamanan produk Anda. Ikuti `tcp, #, #, 0.0.0.0/0` formatnya. Misalnya, gunakan `tcp, 22, 22, 0.0.0.0/0` untuk SSH dan `tcp, 3389, 3389, 0.0.0.0/0` untuk RDP.
 10. Dalam Topologi Kluster dan AWS Sumber Daya 1: Judul ke Kluster dan AWS Sumber Daya Topologi 3: Diagram Arsitektur kolom URL, masukkan data untuk produk Anda. CloudFormation Anda harus memasukkan data berikut:

- Judul topologi — Judul opsi penerapan atau pemenuhan Anda. Judul muncul di halaman detail produk Anda di bagian Opsi Pemenuhan. Misalnya:



- Di kolom Perkiraan Harga, masukkan tautan ke [Kalkulator AWS](#) dengan nilai Anda.
- Deskripsi Pendek dan Panjang — Masukkan deskripsi opsi penerapan Anda.
- URL Template - Berikan tautan yang dapat diunduh ke template Cloudformation Anda.

- Diagram Arsitektur — Berikan tautan yang dapat diunduh ke diagram arsitektur topologi Anda CloudFormation. Setiap opsi Deployment harus memiliki diagram berbeda yang menunjukkan apa yang diluncurkan stack. Diagram harus mengikuti persyaratan yang tercantum di [Diagram arsitektur](#)

Mengirimkan PLF

Langkah-langkah berikut menjelaskan cara mengirimkan PLF yang telah selesai.

1. Masuk ke [Portal Manajemen AWS Marketplace](#).
2. Pada Asettab, pilih Upload File.
3. Pada halaman Unggahan File, unggah PLF Anda dan templat apa pun AWS CloudFormation . Pengunggah file menyediakan mekanisme transfer yang aman dan riwayat file yang dikirimkan. Pengunggah secara otomatis memberi tahu AWS Marketplace, yang meninjau kiriman Anda untuk kepatuhan kebijakan dan keamanan, kerentanan perangkat lunak, dan kegunaan produk. Jika tim memiliki pertanyaan atau masalah dengan permintaan, mereka mengirimi Anda pesan email.

Memperbarui produk Anda

Untuk produk yang Anda buat dengan menggunakan product load form (PLF), Anda juga menggunakan PLF untuk melakukan perubahan pada produk tersebut. Anda dapat membuat perubahan pada PLF asli yang Anda selesaikan atau, jika tidak tersedia, Anda dapat memulai dengan PLF baru. Sama seperti menggunakan tab Produk, Anda dapat menambahkan versi baru, menghapus versi yang ada, dan memperbarui harga, jenis instans, ketersediaan Wilayah, dan metadata. Untuk melakukan pembaruan, Anda menyiapkan produk yang diperbarui dengan cara yang sama seperti Anda menyiapkan produk baru. Setelah pembaruan produk disiapkan, ikuti langkah-langkah ini:

1. Gunakan PLF yang ada, atau mulai [Portal Manajemen AWS Marketplace](#), dan pada tab Aset, pilih Unggah file. Di bawah formulir beban Produk dan panduan penjual, Anda dapat mengunduh PLF untuk produk Anda.
2. Perbarui produk di PLF.
3. Dari [Portal Manajemen AWS Marketplace](#), di bawah tab Aset, pilih Unggah File.
4. Pada halaman Unggahan File, unggah PLF Anda yang diperbarui dan templat apa pun AWS CloudFormation . Pengunggah file menyediakan mekanisme transfer yang aman dan riwayat file

yang dikirimkan. Pengunggah secara otomatis memberi tahu AWS Marketplace tim untuk mulai memproses permintaan Anda. Sertakan deskripsi kiriman (menambahkan versi baru, mengubah harga, mengubah metadata, dan sebagainya).

Pengiriman produk Anda ditinjau untuk kepatuhan kebijakan dan keamanan, kerentanan perangkat lunak, dan kegunaan produk. Jika ada pertanyaan atau masalah dengan permintaan, AWS Marketplace tim akan menghubungi Anda melalui pesan email. Pembaruan ke halaman produk yang ada diproses dan dirilis secara langsung tanpa ulasan tambahan.

Perubahan dan pembaruan produk

Penjual dapat mengirimkan perubahan pada produk mereka kapan saja, dan mereka akan diproses seperti yang dijelaskan sebelumnya. Namun, beberapa perubahan hanya dapat dilakukan setiap 90 atau 120 hari, atau ketika perubahan yang tertunda dilakukan. Contohnya termasuk perubahan harga dan Wilayah AWS atau perubahan jenis instance. Perubahan umum meliputi:

- **Versi Baru** — Versi baru dari perangkat lunak dan peluncuran tambalan atau pembaruan. Atas permintaan Anda, kami dapat memberi tahu pelanggan yang telah berlangganan AWS Marketplace konten Anda tentang ketersediaan versi baru atau mengirim instruksi peningkatan atas nama Anda.
- **Perubahan metadata** — Perubahan pada informasi produk (Deskripsi URLs, dan Petunjuk Penggunaan).
- **Perubahan Harga** — Perubahan pada jumlah harga. Pemberitahuan kepada pelanggan saat ini dikirim setelah permintaan selesai. Setelah pemberitahuan dikirim, perubahan harga akan berlaku pada bulan pertama setelah jendela 90 hari. Misalnya, jika Anda membuat perubahan pada 16 Maret, 90 hari setelahnya akan menjadi sekitar 16 Juni, tetapi perubahan harga terjadi pada bulan pertama berikutnya. Tanggal sebenarnya dari perubahan tersebut adalah 1 Juli.
- **Perubahan Model Harga** — Perubahan pada model penetapan harga (misalnya, Per Jam, Gratis, Hourly_Annual). Tidak semua perubahan model harga didukung, dan semua permintaan untuk mengubah model harus ditinjau dan disetujui oleh AWS Marketplace tim. Setiap perubahan dari model gratis ke model berbayar menghadirkan dampak signifikan bagi pelanggan yang sudah ada. Alternatifnya adalah mengusulkan produk baru dengan fitur tambahan dan mendorong pelanggan saat ini untuk bermigrasi.
- **Perubahan Wilayah atau Instance** - Menambahkan atau menghapus jenis atau Wilayah instance.
- **Penghapusan produk** — Hapus halaman produk AWS Marketplace untuk mencegah pelanggan baru berlangganan. Pemberitahuan kepada pelanggan saat ini dikirim setelah permintaan selesai.

Waktu dan harapan

Meskipun kami berusaha untuk memproses permintaan secepat mungkin, permintaan dapat memerlukan beberapa iterasi dan peninjauan oleh penjual dan AWS Marketplace tim. Gunakan berikut ini sebagai panduan untuk waktu yang dibutuhkan untuk menyelesaikan proses:

- Total waktu permintaan biasanya memakan waktu 2-4 minggu kalender. Permintaan atau produk yang lebih kompleks dapat memakan waktu lebih lama, karena beberapa iterasi dan penyesuaian metadata produk dan perangkat lunak.
- Kami memerlukan permintaan produk lengkap dan AMI setidaknya 45 hari sebelum acara atau rilis yang direncanakan, sehingga kami dapat memprioritaskan permintaan yang sesuai.

Jika Anda memiliki pertanyaan tentang permintaan Anda, hubungi tim [Operasi AWS Marketplace Penjual](#).

AMIs Mengirimkan ke AWS Marketplace

Semua yang AMIs dibangun dan diserahkan AWS Marketplace harus mematuhi semua kebijakan produk. Kami menyarankan beberapa pemeriksaan akhir AMI Anda sebelum pengiriman:

- Hapus semua kredensi pengguna dari sistem; misalnya, semua kata sandi default, kunci otorisasi, pasangan kunci, kunci keamanan, atau kredensi lainnya.
- Pastikan bahwa login root dinonaktifkan atau dikunci. Hanya akun akses sudo yang diizinkan.
- Jika Anda mengirimkan AMI untuk diterapkan ke AWS GovCloud (US) Wilayah, Anda harus [memiliki GovCloud akun AWS aktif dan menyetujui Persyaratan AWS](#), termasuk [GovCloud persyaratan kontrol](#) ekspor yang berlaku.

Pemindaian swalayan AMI

AWS Marketplace menawarkan pemindaian AMI di dalam Portal Manajemen AWS Marketplace. Untuk produk baru, sistem secara otomatis menjalankan pemindaian dan memberikan hasil setelah pengiriman. Untuk versi baru produk yang ada, gunakan fitur Uji 'Tambah Versi' untuk memulai pemindaian. Proses ini biasanya selesai dalam waktu kurang dari satu jam dan memberikan umpan balik yang komprehensif di satu lokasi yang nyaman.

Untuk mulai berbagi dan memindai AMI Anda dengan pemindaian swalayan

1. Arahkan ke <https://aws.amazon.com/marketplace/management/products/server>.
2. Di kolom Judul produk, pilih produk.
3. Pada halaman detail produk, di bagian Versi, pilih Uji 'Tambahkan Versi'.
4. Pada halaman Uji 'Tambahkan Versi', masukkan informasi versi dan parameter validasi yang diperlukan.
5. Memilih Menjalankan pengujian.
6. Lihat hasil pemindaian.

Setelah AMI Anda berhasil dipindai, Anda dapat mengikuti proses saat ini untuk mengirimkannya ke tim Operasi AWS Marketplace Penjual dengan [mengunggah formulir pemuatan](#) produk (PLF) Anda. Jika Anda memiliki masalah, hubungi tim [Operasi AWS Marketplace Penjual](#).

Untuk memasukkan AMI Anda dalam daftar pemindaian swalayan, AMI harus berada di Wilayah us-east-1 (Virginia N.) dan dimiliki oleh akun AWS Marketplace penjual Anda. Jika Anda perlu memberikan akses akun lain ke akun Portal Manajemen AWS Marketplace, Anda harus mendaftarkan akun tersebut sebagai penjual. Untuk informasi selengkapnya, lihat [Mendaftar sebagai AWS Marketplace penjual](#).

Kloning AMI dan penetapan kode produk

Setelah AMI Anda dikirimkan, AWS Marketplace buat kloning AMIs untuk setiap Wilayah yang telah Anda indikasikan bahwa perangkat lunak harus tersedia. Selama proses kloning dan penerbitan ini, AWS Marketplace lampirkan kode produk ke kloning. AMIs Kode produk digunakan untuk mengontrol akses dan penggunaan meter. Semua kiriman harus melalui proses kloning AMI ini.

Daftar periksa akhir

Untuk membantu menghindari keterlambatan dalam mempublikasikan produk Anda, gunakan daftar periksa ini sebelum Anda mengirimkan permintaan produk Anda.

Penggunaan produk

- Siap produksi.
- Tidak membatasi penggunaan produk berdasarkan waktu atau batasan lainnya.
- Kompatibel dengan pengalaman pemenuhan 1-klik.

- Perangkat lunak ini berisi semua yang diperlukan untuk menggunakan produk, termasuk aplikasi klien.
- Pengguna default menggunakan kata sandi acak dan/atau pembuatan pengguna awal memerlukan verifikasi bahwa pembeli berwenang untuk menggunakan instance menggunakan nilai yang unik untuk instance seperti ID instance.

Untuk produk gratis atau berbayar

- Tidak ada lisensi tambahan yang diperlukan untuk menggunakan produk.
- Perangkat lunak yang dikemas ulang berbayar memenuhi. AWS Marketplace [Persyaratan untuk mengirimkan perangkat lunak paket ulang berbayar](#)
- Pembeli tidak harus memberikan informasi yang dapat diidentifikasi secara pribadi (misalnya, alamat email) untuk menggunakan produk.

Persiapan AMI

- Gunakan virtualisasi hardware virtual machine (HVM) dan arsitektur 64-bit.
- Tidak mengandung kerentanan, malware, atau virus yang diketahui.
- Pembeli memiliki akses administrasi tingkat sistem operasi ke AMI.
- Gunakan Uji 'Tambahkan Versi' di bagian Portal Manajemen AWS Marketplace untuk memindai AMI Anda.

Untuk Windows AMIs

- Gunakan versi terbaru `Ec2ConfigService`, seperti yang dijelaskan dalam [Mengkonfigurasi Windows contoh menggunakan EC2Config layanan](#) di Panduan EC2 Pengguna Amazon.
- `Ec2HandleUserDataPlugin Ec2SetPasswordEc2WindowsActivate`, dan diaktifkan, seperti yang dijelaskan dalam [Mengkonfigurasi Windows contoh menggunakan EC2Config layanan](#) di Panduan EC2 Pengguna Amazon.
- Tidak ada Akun Tamu atau Pengguna Desktop Jarak Jauh yang hadir.

Untuk Linux AMIs

- Login root dikunci dan dinonaktifkan.
- Tidak ada kunci resmi, kata sandi default, atau kredensi lainnya yang disertakan.

- Semua bidang yang diperlukan selesai.
- Semua nilai berada dalam batas karakter yang ditentukan.
- Semua URLs beban tanpa kesalahan.
- Gambar produk setidaknya 110px lebar dan antara rasio 1:1 dan 2:1.
- Harga ditentukan untuk semua jenis instans yang diaktifkan (untuk model harga per jam, hourly_monthly, dan hourly_annual).
- Harga bulanan ditentukan (untuk model harga per jam_bulanan dan bulanan).

Jika Anda memiliki pertanyaan atau komentar tentang pembuatan AMI otomatis, hubungi tim [Operasi AWS Marketplace Penjual](#).

Memasarkan produk Anda di AWS Marketplace

Anda dapat berkontribusi pada keberhasilan produk Anda dengan mendorong kesadaran AWS Marketplace dan dengan mengarahkan lalu lintas langsung ke halaman produk Anda AWS Marketplace. Bagian berikut memberikan informasi dan dukungan untuk membantu Anda memasarkan produk atau produk yang telah Anda daftarkan AWS Marketplace.

Topik

- [Akademi GTM 180 hari](#)
- [Mengumumkan ketersediaan produk Anda](#)
- [AWS Marketplace pesan](#)
- [Ulasan tentang AWS Marketplace](#)
- [Menautkan ke AWS Marketplace](#)
- [Menggunakan AWS Marketplace branding](#)
- [Memahami Beli dengan AWS sebagai penjual](#)
- [Menautkan langsung ke produk Anda di AWS Marketplace](#)
- [Siaran pers](#)
- [AWS Marketplace pedoman penggunaan merek dagang](#)

Akademi GTM 180 hari

Akademi GTM 180 hari tersedia untuk semua AWS Marketplace penjual, dan menyediakan sumber daya swalayan go-to-market (GTM) untuk membantu Anda membangun, mengaktifkan, dan melacak kampanye pembuatan permintaan untuk penawaran Anda. AWS Marketplace Anda dapat:

- Permintaan jalur cepat untuk daftar Anda.
- Tingkatkan laba atas investasi pemasaran dan tingkatkan pesan pelanggan dengan mengintegrasikan AWS dan AWS Marketplace menilai proposisi ke dalam pesan Anda.
- Anda dapat maju menuju atau dalam Panduan [Program AWS Marketplace Go-to-Market](#).

Anda dapat mengakses Akademi GTM 180 hari dari bagian Sumber Daya Marketplace di bagian. [Portal Manajemen AWS Marketplace](#)

Mengumumkan ketersediaan produk Anda

Kami mendorong Anda untuk secara luas mengumumkan ketersediaan produk Anda di AWS Marketplace. Anda dapat melakukan ini melalui siaran pers, tweet, blog, atau saluran media pilihan lainnya. Kami telah menyediakan contoh teks yang dapat Anda sertakan, bersama dengan pedoman dan instruksi untuk menggunakan merek dagang kami dan mengeluarkan siaran pers.

Kami akan meninjau blog, tweet, dan pengumuman non-press release Anda sebelum go public untuk memastikan konsistensi dengan AWS pesan dan pedoman merek atau suara. Kirimkan permintaan Anda untuk ditinjau ke manajer AWS akun Anda. Tinjauan ini membutuhkan waktu hingga 10 hari kerja untuk diselesaikan. Beri tahu kami ketika Anda memposting tweet, blog, atau siaran pers apa pun, dan kami akan melakukan yang terbaik untuk memposting ulang untuk meningkatkan visibilitas mereka.

AWS Marketplace pesan

Dalam komunikasi pelanggan Anda, Anda mungkin ingin menggambarkan tujuan, sasaran, dan manfaat dari pembelian produk Anda menggunakan AWS Marketplace. Gunakan pesan berikut saat merujuk ke AWS Marketplace.

Apa itu AWS Marketplace?

AWS Marketplace adalah toko online yang memudahkan pelanggan untuk menemukan, membandingkan, dan segera mulai menggunakan perangkat lunak dan layanan yang berjalan di AWS. Pengunjung AWS Marketplace dapat menggunakan penyebaran 1-Klik untuk meluncurkan perangkat lunak yang telah dikonfigurasi dengan cepat dan hanya membayar untuk apa yang mereka gunakan, per jam atau bulan. AWS menangani penagihan dan pembayaran, dan biaya perangkat lunak muncul di AWS tagihan pelanggan.

Mengapa pelanggan berbelanja AWS Marketplace?

Menemukan dan menerapkan perangkat lunak dapat menjadi tantangan. AWS Marketplace fitur berbagai pilihan perangkat lunak TI dan bisnis komersial dan gratis, termasuk infrastruktur perangkat lunak seperti database dan server aplikasi, solusi IoT, alat pengembang, dan aplikasi bisnis, dari penjual populer. AWS Marketplace memungkinkan pelanggan untuk membandingkan opsi, membaca ulasan, dan dengan cepat menemukan perangkat lunak yang mereka inginkan. Kemudian mereka dapat menerapkannya ke instans Amazon Elastic Compute Cloud mereka sendiri menggunakan 1-Klik atau menggunakan file. Portal Manajemen AWS Marketplace

Harga perangkat lunak diposting dengan jelas di situs web dan pelanggan dapat membeli sebagian besar perangkat lunak segera, dengan instrumen pembayaran yang sudah ada di file dengan Amazon Web Services. Biaya perangkat lunak muncul pada tagihan bulanan yang sama dengan biaya AWS infrastruktur.

Mengapa perangkat lunak atau penjual SaaS menjual? AWS Marketplace

Dengan AWS Marketplace, perangkat lunak dan perangkat lunak sebagai layanan (SaaS) penjual dengan penawaran yang berjalan AWS dapat memperoleh manfaat dari peningkatan kesadaran pelanggan, penyebaran yang disederhanakan, dan penagihan otomatis.

AWS Marketplace membantu penjual perangkat lunak dan SaaS perangkat lunak dan layanan yang berjalan AWS mencari pelanggan baru dengan mengekspos produk mereka ke beberapa dari ratusan ribu AWS pelanggan, mulai dari pengembang perangkat lunak individu hingga perusahaan besar.

Menjual AWS Marketplace memungkinkan vendor perangkat lunak independen (ISVs) untuk menambahkan tagihan per jam untuk perangkat lunak mereka tanpa melakukan perubahan kode yang mahal. Mereka hanya mengunggah Amazon Machine Image (AMI) ke AWS dan memberikan biaya per jam. Penagihan dikelola oleh AWS Marketplace, membebaskan penjual dari tanggung jawab pengukuran penggunaan, mengelola akun pelanggan, dan memproses pembayaran, membuat pengembang perangkat lunak lebih banyak waktu untuk fokus membangun perangkat lunak yang hebat.

Selain itu, pelanggan mendapat manfaat dari kemampuan untuk dengan mudah menyebarkan gambar perangkat lunak yang telah dikonfigurasi sebelumnya, menyederhanakan orientasi untuk pelanggan baru.

Ulasan tentang AWS Marketplace

AWS Marketplace memberikan kemampuan bagi pelanggan untuk mengirimkan ulasan tentang produk Anda. Kami juga menyediakan kemampuan untuk pengulas sindikasi seperti G2, business-to-business pasar yang mengkurasi ulasan produk independen, untuk mengintegrasikan ulasan sindikasi mereka. AWS Marketplace

AWS Marketplace ulasan pelanggan harus memenuhi pedoman ulasan yang tercantum dalam panduan pengguna untuk pembeli. Kiriman ulasan tidak dirilis AWS Marketplace sampai setelah kiriman ditinjau untuk memverifikasinya memenuhi kriteria peninjauan kami. Untuk informasi

selengkapnya tentang pedoman peninjauan, lihat [Ulasan Produk](#). Organisasi peninjau sindikasi menggunakan proses unik mereka sendiri untuk memvalidasi ulasan mereka dan tidak ditinjau oleh AWS Marketplace sebelum rilis. Jika Anda berpikir bahwa tinjauan sindikasi pada produk Anda tidak memenuhi pedoman peninjauan produk, atau ulasan pada produk Anda berisi konten yang tidak menyenangkan, hubungi tim Operasi [AWS Marketplace Penjual](#).

Peninjau juga dapat memberikan peringkat bintang untuk produk Anda berdasarkan sistem peringkat bintang lima. Peringkat dirata-ratakan untuk memberikan peringkat bintang keseluruhan untuk produk Anda. Ulasan sindikasi juga dapat mencakup peringkat bintang, tetapi peringkat bintang dari ulasan sindikasi tidak dirata-ratakan dengan peringkat bintang AWS pelanggan.

Berikut ini adalah poin-poin penting tambahan tentang fitur review produk:

- Anda tidak dapat menghapus ulasan produk AWS Marketplace. Namun, Anda dapat meninggalkan komentar pada ulasan apa pun selama komentar tersebut memenuhi kriteria ulasan yang mengatur ulasan produk.
- Jika menurut Anda ulasan tidak memenuhi pedoman peninjauan atau berisi konten yang tidak menyenangkan, Anda dapat menghubungi tim [Operasi AWS Marketplace Penjual](#) dan menjelaskan kekhawatiran Anda.
- AWS pelanggan yang mencari produk di AWS Marketplace dapat mencari dan memfilter hasil berdasarkan peringkat, ulasan terverifikasi, dan ulasan yang bersumber dari luar. AWS pelanggan melihat peringkat yang bersumber secara eksternal di samping peringkat AWS pelanggan di hasil pencarian.
- Ulasan sindikasi untuk produk Anda ditambahkan secara otomatis tanpa AWS Marketplace biaya kepada Anda. Karena ulasan ditambahkan secara otomatis, Anda tidak perlu mengirimkan permintaan untuk menambahkan tinjauan sindikasi.
- Jika Anda tidak memiliki ulasan sindikasi untuk produk Anda, Anda dapat menghubungi pengulas sindikasi dan mengikuti proses mereka untuk mendapatkan produk Anda ditinjau. Misalnya, dengan G2, Anda dapat mengunjungi situs web mereka dan mengklaim halaman produk Anda untuk memulai proses peninjauan mereka.

Menautkan ke AWS Marketplace

Perusahaan Anda mungkin memiliki kehadiran web di mana ia menggambarkan dan mempromosikan produk Anda. Kami mendorong Anda untuk menyoroti bahwa produk tersedia untuk dijalankan AWS dan dapat dibeli menggunakan AWS Marketplace. Untuk menyederhanakan

proses bagi pelanggan Anda untuk menemukan dan menyebarkan perangkat lunak Anda, kami telah memberikan instruksi untuk menghubungkan pelanggan Anda ke produk Anda.

Menggunakan AWS Marketplace branding

Note

Anda harus menjadi AWS Mitra untuk menggunakan panduan dan aset yang tercantum di sini. Untuk informasi tentang menjadi AWS Mitra, lihat [Bergabung dengan Jaringan AWS Mitra](#).

AWS Marketplace menyediakan panduan pesan dan aset branding untuk membantu Anda mempublikasikan perangkat lunak, data, atau layanan Anda. Jika Anda ingin mempromosikan iklan Anda AWS Marketplace, Anda dapat mengunduh sumber daya berikut:

- AWS Panduan Kreatif dan Pesan Mitra 2023
- Tersedia dalam aset AWS Marketplace branding

Untuk mengunduh sumber daya

1. Masuk ke [AWS Partner Central](#).
2. Rekatkan URL berikut ke bilah alamat browser Anda dan buka halaman: [Panduan Kreatif dan Pesan AWS Mitra 2023](#).
3. Pilih Unduh.
4. Ulangi langkah 2 dan 3 untuk [tersedia dalam](#) aset AWS Marketplace branding.

Memahami Beli dengan AWS sebagai penjual

Beli dengan AWS memungkinkan AWS Marketplace penjual untuk menampilkan dan menjual AWS Marketplace produk di situs web mereka sendiri. AWS Marketplace penjual dapat berintegrasi dengan tombol AWS Marketplace call-to-action (CTA) dan APIs untuk membangun penemuan perangkat lunak yang AWS didukung dan pengalaman pengadaan di situs web milik penjual. Ini termasuk properti web yang dimiliki oleh AWS penjual, termasuk vendor perangkat lunak independen (), Mitra Saluran (ISVs), Distributor (CPs). DPs

 Note

Untuk informasi selengkapnya tentang Beli dengan AWS fitur, lihat [Beli dengan AWS](#).

Tombol CTA meliputi:

- Beli dengan AWS
- Lihat penawaran di AWS
- Coba gratis dengan AWS
- Minta penawaran pribadi
- Minta demo

Beli dengan AWS dapat memberikan manfaat sebagai berikut:

- AWS co-branding dan penemuan di situs web Anda: Buat pengalaman co-branded di situs web Anda yang didukung oleh AWS Marketplace. Memungkinkan pelanggan untuk mencari daftar produk yang dikuratori dari AWS Marketplace situs web Anda, memfilter produk, mengakses ulasan, meminta harga khusus, dan melakukan pembelian dengan percaya diri menggunakan AWS akun mereka.
- Kecepatan pengadaan untuk AWS pelanggan: AWS Tombol Beli dengan di situs web Anda memberi pelanggan kemampuan untuk berlangganan uji coba gratis dengan cepat atau melakukan pembelian menggunakan akun mereka AWS. Beli dengan AWS memberikan pengalaman pembelian merek bersama yang efisien kepada pelanggan melalui AWS Marketplace.
- Manfaatkan operasi yang ada di AWS Marketplace: Perluas manfaat penemuan dan pengadaan AWS Marketplace kepada pelanggan melalui situs web Anda, sambil memanfaatkan proses yang ada AWS Marketplace untuk mengelola daftar, pencairan, dukungan, dan wawasan Anda, tanpa upaya tambahan.

Beli dengan AWS prasyarat

Untuk mengaktifkan Beli dengan AWS, Anda harus:

- [Terdaftar sebagai penjual](#) di AWS Marketplace.
- Jadilah penjual atau pengecer resmi AWS Marketplace dan menyetujui [Ketentuan Layanan untuk AWS Marketplace Penjual](#).

- Patuhi [pedoman AWS merek dagang](#).
- Patuhi [Lisensi Beli dengan AWS Merek Dagang](#).
- Miliki setidaknya satu daftar produk SaaS publik di AWS Marketplace

 Note

Saat ini, Beli dengan hanya AWS mendukung produk SaaS.

- Patuhi pedoman Beli dengan AWS branding:
 - Masuk ke akun [AWS Partner Central](#) Anda untuk mengunduh toolkit pemasaran yang berisi aset branding dan spesifikasi yang harus diikuti untuk elemen AWS Marketplace branding dan aset pemasaran.
 - Lihat [panduan Beli dengan AWS kreatif dan pesan](#) untuk petunjuk tentang penggunaan elemen branding dan aset pemasaran.

[Beli dengan AWS: Memulai sebagai penjual](#)

Tinjau perubahan branding

Selesaikan langkah-langkah berikut untuk mengajukan permintaan tinjauan merek sebelum diluncurkan dengan Beli dengan AWS pengalaman:

1. Arahkan ke Portal Manajemen AWS Marketplace dan [kirimkan tiket dukungan](#) untuk Commercial Marketplace
 - a. Pilih Beli AWS dengan kategori
 - b. Pilih Branding Review sebagai sub-kategori.
2. Berikan yang berikut di tiket:
 - a. Alamat email utama Anda, dan ID penjual.
 - b. Di bagian Berikan detail lebih lanjut tentang permintaan Anda:
 - i. URL tempat Anda akan mengimplementasikan Beli dengan AWS integrasi
 - ii. Tautan ke kotak pasir, situs langsung, atau maket Anda untuk ditinjau oleh tim AWS Marketplace operasi. Jika Anda tidak menyertakan ini, tim operasi akan memintanya setelah Anda mengirimkan tiket.

3. Tim operasi AWS Marketplace penjual akan mengevaluasi implementasi Anda terhadap Buy dengan pedoman AWS branding. Mereka akan memberikan umpan balik untuk memastikan kepatuhan. Anda harus menangani semua umpan balik dan mendapatkan persetujuan sebelum meluncurkan.

Beli dengan AWS pelaporan

Beli dengan AWS metrik menawarkan penjual tampilan komprehensif dari corong konversi mereka. Dasbor Beli dengan AWS penjual memberikan wawasan terperinci untuk mengukur keterlibatan pengguna. Penjual dapat mengakses Beli dengan AWS metrik melalui [Portal Manajemen AWS Marketplace](#)

Untuk melihat dasbor Beli dengan AWS metrik:

1. Masuk ke [Portal Manajemen AWS Marketplace](#).
2. Arahkan ke Wawasan.
3. Klik Pemasaran.
4. Klik Beli dengan AWS

Atau, penjual yang telah menautkan akun mereka dapat mengakses metrik melalui AWS Partner Central. Untuk melihat metrik ini, masuk ke [AWS Partner Central](#) dan navigasikan ke bagian Analytics.

Menautkan langsung ke produk Anda di AWS Marketplace

Tambahkan tautan dalam ke situs web atau materi pemasaran Anda untuk mengarahkan pelanggan ke halaman produk tertentu AWS Marketplace. Gunakan format tautan berikut:

```
https://aws.amazon.com/marketplace/pp/prodview-prodview ID
```

Ganti *prodview ID* dengan kode alfanumerik unik untuk produk Anda. AWS Marketplace Misalnya:

Example

```
https://aws.amazon.com/marketplace/pp/prodview-1234567890abcdef
```

[ID prodview muncul di URL saat Anda mencari aplikasi di aws.amazon.com/marketplace.](https://aws.amazon.com/marketplace) Untuk bantuan menemukan ID prodview Anda, hubungi manajer akun Anda.

 Note

Uji tautan sebelum menggunakannya untuk memastikan bahwa mereka mengarahkan pelanggan Anda ke halaman yang benar.

Siaran pers

Kami mendorong Anda untuk mengumumkan ketersediaan produk Anda AWS Marketplace melalui saluran apa pun yang Anda inginkan. Namun, semua siaran pers yang referensi AWS Marketplace harus ditinjau dan ditandatangani oleh Amazon sebelum publikasi atau pengumuman dibuat. Meskipun kami mendorong Anda untuk membuat pengumuman, kami tidak dapat mendukung siaran pers bersama dengan AWS Marketplace penjual. Kami akan, atas case-by-case dasar, mendukung siaran pers dengan kutipan dari AWS. Kutipan harus memenuhi beberapa persyaratan, termasuk namun tidak terbatas pada: ia mengumumkan produk atau layanan baru yang terdaftar di AWS Marketplace atau termasuk referensi pelanggan yang menggunakan AWS Marketplace.

Anda harus menyusun semua siaran pers. Kami menyarankan judul berikut: [Masukkan nama produk] Sekarang Tersedia di AWS Marketplace. Gunakan pesan dalam dokumen ini untuk konsistensi.

Siaran pers harus:

- Jelaskan dengan jelas dan akurat bagaimana pengumuman tersebut berhubungan dengan Amazon.com
- Memperjelas peran Anda pada AWS dan dengan pelanggan
- Berfokus pada pelanggan dan menekankan manfaat pelanggan

Siaran pers tidak boleh:

- Gunakan istilah mitra, kemitraan, atau aliansi untuk menggambarkan hubungan. Kami lebih suka kesepakatan, kerja sama, atau hubungan.
- Sertakan kutipan dari eksekutif Amazon Web Services kecuali sebelumnya disepakati.
- Sertakan proyeksi penjualan atau gunakan.com oleh pedagang kecuali merujuk ke situs web di boilerplate perusahaan Anda.

- Merujuk ke organisasi Anda sebagai rekanan Amazon.com karena ini dapat dikacaukan dengan Amazon Associates, program afiliasi online kami.
- Mengungkapkan informasi kepemilikan tentang Amazon.com atau lihat simbol ticker saham kami.

Minta siaran pers Anda ditinjau dengan mengirimkannya dalam format teks ke manajer akun Anda. Selain itu, tinjau [pedoman merek dagang Amazon Web Services](#) sebelum menggunakan AWS merek dagang apa pun. Pedoman khusus untuk AWS Marketplace merek dagang ada di bagian berikut.

AWS Marketplace pedoman penggunaan merek dagang

Pedoman ini berlaku untuk penggunaan AWS Marketplace logo dan merek dagang Anda, (masing-masing “Merek Dagang” dan secara kolektif disebut “Merek Dagang”) dalam materi yang telah disetujui sebelumnya oleh Amazon.com, Inc. dan/atau afiliasinya (“Amazon”). Kepatuhan yang ketat terhadap Pedoman ini diperlukan setiap saat, dan setiap penggunaan Merek Dagang yang melanggar Pedoman ini akan secara otomatis mengakhiri lisensi apa pun yang terkait dengan penggunaan Merek Dagang oleh Anda.

1. Anda dapat menggunakan Merek Dagang semata-mata untuk tujuan yang secara tegas diizinkan oleh Amazon dan penggunaan Anda harus: (i) mematuhi sebagian besar up-to-date versi dari semua perjanjian dengan Amazon mengenai penggunaan Merek Dagang (secara kolektif disebut “Perjanjian”); (ii) mematuhi sebagian besar up-to-date versi Pedoman ini; dan (iii) mematuhi syarat, ketentuan, atau kebijakan lain yang dapat dikeluarkan Amazon dari waktu ke waktu yang berlaku untuk penggunaan Merek Dagang.
2. Kami akan menyediakan gambar Merek Dagang yang disetujui untuk Anda gunakan. Anda tidak boleh mengubah Merek Dagang dengan cara apa pun, termasuk namun tidak terbatas pada, mengubah proporsi, warna, atau font Merek Dagang, atau menambahkan atau menghapus elemen apa pun dari Merek Dagang.
3. Anda tidak boleh menggunakan Merek Dagang dengan cara apa pun yang menyiratkan sponsor atau dukungan oleh Amazon selain dengan menggunakan Merek Dagang sebagaimana secara khusus diizinkan berdasarkan Perjanjian.
4. Anda tidak boleh menggunakan Merek Dagang untuk meremehkan Amazon, produk atau layanannya, atau dengan cara yang, atas kebijakan Amazon sendiri, dapat mengurangi atau merusak atau menodai niat baik Amazon dalam Merek Dagang.
5. Merek Dagang harus muncul dengan sendirinya, dengan jarak yang wajar antara setiap sisi Merek Dagang dan elemen visual, grafik, atau tekstual lainnya. Dalam keadaan apa pun Merek

Dagang tidak boleh ditempatkan pada latar belakang apa pun yang mengganggu keterbacaan atau tampilan Merek Dagang.

6. Anda harus menyertakan pernyataan berikut dalam materi apa pun yang menampilkan Merek Dagang: “AWS Marketplace dan AWS Marketplace logo adalah merek dagang dari Amazon.com, Inc. atau afiliasinya.
7. Anda mengakui bahwa semua hak atas Merek Dagang adalah milik eksklusif Amazon, dan semua niat baik yang dihasilkan melalui penggunaan Merek Dagang oleh Anda akan berlaku untuk kepentingan Amazon. Anda tidak akan mengambil tindakan apa pun yang bertentangan dengan hak Amazon dalam, atau kepemilikan, Merek Dagang.

Amazon berhak, dapat dilaksanakan atas kebijakannya sendiri, untuk mengubah Pedoman ini dan/atau Merek Dagang yang disetujui kapan saja dan untuk mengambil tindakan yang tepat terhadap penggunaan apa pun tanpa izin atau penggunaan apa pun yang tidak sesuai dengan Pedoman ini. Jika Anda memiliki pertanyaan tentang Pedoman ini, hubungi trademarks@amazon.com untuk bantuan atau kirimkan surat kepada kami di alamat berikut:

Amazon.com, Inc., Perhatian: Merek Dagang

Kotak PO 81226

Seattle, WA 98108-1226

Pemberitahuan penjual untuk AWS Marketplace acara

Anda dapat menerima pemberitahuan tepat waktu terkait AWS Marketplace produk Anda melalui email, EventBridge acara Amazon, dan topik Simple Notification Service Amazon (Amazon SNS). AWS Marketplace menyediakan. Misalnya, Anda dapat menerima pemberitahuan terkait penawaran pribadi, permintaan produk layanan profesional, dan pengingat pemindaian berulang. Topik ini memberikan ikhtisar opsi pemberitahuan dan acara yang tersedia.

- **Pemberitahuan email** — Di dalam AWS Marketplace, penjual, pembeli, dan vendor perangkat lunak independen (ISVs) dapat menerima pemberitahuan email. AWS Marketplace menggunakan akun root Anda untuk mengirim email otomatis secara real time ke email AWS akun Anda, memperbarui Anda pada AWS Marketplace penawaran dan perjanjian. Anda juga dapat menambahkan alias email khusus untuk notifikasi dan berhenti berlangganan penerima dari notifikasi email. Untuk informasi selengkapnya, lihat [Mengelola notifikasi email untuk AWS Marketplace acara](#).
- **Topik Amazon SNS** — Untuk menerima pemberitahuan tentang perubahan langganan pelanggan untuk produk Anda, Anda dapat berlangganan topik Amazon SNS untuk AWS Marketplace diberikan kepada Anda selama pembuatan produk. Misalnya, Anda dapat mengetahui kapan pelanggan menerima penawaran pribadi. Untuk selengkapnya, lihat topik Amazon SNS untuk jenis produk Anda:
 - [Perangkat lunak sebagai layanan \(SaaS\) produk](#)
 - [Produk Gambar Mesin Amazon \(AMI\)](#)
 - [Produk kontainer](#)
- **EventBridge**— Anda dapat menggunakan EventBridge untuk menerima acara untuk tindakan yang terjadi di AWS Marketplace, misalnya, ketika penawaran dibuat. Acara ini berisi informasi seperti ID, tanggal kedaluwarsa, dan detail produk. Untuk informasi selengkapnya, lihat [EventBridge Acara Amazon](#) dan [Panduan EventBridge Pengguna Amazon](#).

Topik berikut memberikan informasi lebih lanjut tentang pemberitahuan dan acara di AWS Marketplace.

Topik

- [Mengelola notifikasi email untuk AWS Marketplace acara](#)
- [EventBridge Acara Amazon](#)

Mengelola notifikasi email untuk AWS Marketplace acara

AWS Marketplace mengirimkan pemberitahuan email tentang pembaruan penawaran, perjanjian, langganan, produk, keamanan, penagihan dan pembayaran, dan Private Marketplace. Vendor perangkat lunak independen (ISVs), Mitra AWS Marketplace Saluran, dan pelanggan dapat menerima pemberitahuan email. Untuk contoh dan detail pemberitahuan email yang dikirim untuk pembaruan penawaran dan perjanjian, lihat [Jenis peristiwa](#).

AWS Marketplace mengirimkan pemberitahuan email ke alamat email yang terkait dengan [pengguna root](#) AWS akun Anda. Untuk memperbarui alamat email yang terkait dengan AWS akun Anda, lihat [Perbarui kontak utama untuk AWS akun Anda](#). Anda juga dapat [menambahkan alias email khusus untuk notifikasi dan berhenti berlangganan penerima](#) dari notifikasi email.

Note

Jika Anda kehilangan AWS Marketplace email, periksa folder spam Anda atau sesuaikan pengaturan email. Pemberitahuan email dari AWS Marketplace dikirim dari `no-reply@marketplace.aws`. Penyedia seperti Google dan Yahoo dapat memfilter ini. Untuk petunjuk, lihat [Mencegah email yang valid masuk ke Spam \(Google\)](#) atau [Blokir dan buka blokir alamat email di Yahoo Mail](#).

Topik

- [Jenis peristiwa](#)
- [Deskripsi bidang](#)
- [Kelola notifikasi](#)

Jenis peristiwa

Jenis acara berikut didukung oleh pemberitahuan email untuk semua produk dan jenis harga, kecuali untuk produk pembelajaran mesin.

Penawaran

Tabel berikut menunjukkan acara untuk penawaran. Penawaran adalah seperangkat persyaratan untuk penggunaan produk oleh pembeli. Untuk informasi lebih lanjut, lihat [Mempersiapkan penawaran pribadi untuk AWS Marketplace produk Anda](#).

Email	Peristiwa	Penerima	Judul	Bidang
Penawaran Publikasi Pemberitahuan CP CPPO	Publikasi penawaran pribadi Mitra AWS Marketplace Saluran.	Mitra saluran	AWS Marketplace Penawaran pribadi Channel Partner dipublikasikan	ID AWS akun pelanggan, nama ISV, ID AWS Akun ISV, Nama Produk, ID Produk, ID Penawaran, ID Peluang, Tanggal diterbitkan Penawaran, Tanggal kedaluwarsa penawaran
Penawaran Diterbitkan Pemberitahuan ISV CPPO	Publikasi penawaran pribadi Mitra AWS Marketplace Saluran.	ISV	AWS Marketplace Penawaran pribadi Channel Partner dipublikasikan	AWS Akun pelanggan IDs, Mitra saluran, ID AWS akun mitra saluran, Nama Produk, ID Produk, Nama penawaran, ID Penawaran, ID Peluang, Tanggal diterbitkan penawaran, Tanggal kedaluwarsa penawaran, Harga grosir
OfferPublSVNotificationSCP-1.0	Penjual menerbitkan penawaran pribadi Mitra AWS Marketplace Saluran.	ISV atau Mitra AWS Marketplace Saluran	Penawaran pribadi dipublikasikan	AWS Akun Pelanggan IDs, Nama Produk, ID Produk, Nama Penawaran, ID Penawaran, Tanggal Diterbitkan Penawaran, Tanggal Kedaluwarsa Penawaran, Total Nilai Kontrak
Peluang Reseller Kedaluwarsa	AWS Marketplace Otorisasi	AWS Marketplace Mitra Saluran	Penawaran : Lihat otorisasi penjualan	ISV, Nama Produk, ID Produk

Email	Peristiwa	Penerima	Judul	Bidang
	penjualan Mitra Saluran kedaluwarsa.		yang kedaluwarsa	
Peluang Reseller Dicabut	ISV mencabut otorisasi penjualan Mitra AWS Marketplace Saluran.	AWS Marketplace Mitra Saluran	Penawaran : Lihat otorisasi penjualan yang dinonaktifkan	ISV, Nama Produk, ID Produk
Peluang Reseller Dibuat	ISV menciptakan otorisasi penjualan atau peluang untuk Mitra AWS Marketplace Saluran.	AWS Marketplace Mitra Saluran	Penawaran : Buat penawaran pribadi untuk otorisasi penjualan baru	ISV, Nama Produk, ID Produk, Durasi otorisasi penjualan

Perjanjian

ISVs dan Mitra AWS Marketplace Saluran menerima pemberitahuan email ketika pembeli menerima penawaran umum, penawaran pribadi, atau penawaran pribadi Mitra AWS Marketplace Saluran. Perjanjian dibuat ketika penawaran diterima. Pemberitahuan email juga dikirim untuk peristiwa siklus

hidup perjanjian. Ini termasuk dimulainya perjanjian, pembatalan, penggantian, dan kegagalan setelah penerimaan pembeli. Tabel berikut menunjukkan email yang dikirim untuk acara perjanjian. Pemberitahuan email saat ini tersedia untuk jenis produk tertentu. Jenis produk spesifik Anda mungkin belum memiliki fitur ini.

Email	Peristiwa	Penerima	Judul	Bidang
Penawara Diterima — penawara pribadi atau publik	AWS Marketplace pelanggan menerima penawara publik atau pribadi.	ISV	Pelanggan menerima AWS Marketplace tawaran.	Nama perusahaan pelanggan, ID AWS akun Pelanggan, Nama Produk, ID Produk, ID Perjanjian, Tanggal mulai Perjanjian, Tanggal akhir perjanjian, Tanggal penerimaan perjanjian, Jumlah pembelian
Penawara Diterima — Penawara pribadi Mitra AWS Marketplace Saluran.	AWS Marketplace pelanggan menerima penawara pribadi Mitra AWS Marketplace Saluran.	ISV	Pelanggan menerima penawara pribadi Mitra AWS Marketplace Saluran.	Nama perusahaan pelanggan, ID AWS akun Pelanggan, Nama mitra saluran, ID AWS akun mitra Saluran, Nama Produk, ID Produk, Nama Penawaran, ID Penawaran, ID Perjanjian, Tanggal mulai perjanjian, Tanggal akhir perjanjian, Tanggal penerimaan perjanjian, Harga grosir
Penawara Diterima — Penawara pribadi Mitra AWS	AWS Marketplace pelanggan menerima penawara pribadi	AWS Marketplace Mitra Saluran	Pelanggan menerima penawara pribadi Mitra AWS Marketplace	Nama perusahaan pelanggan, ID AWS akun Pelanggan, nama ISV, ID AWS akun ISV, Nama Produk, ID Produk, Nama Penawaran, ID Penawaran, ID Perjanjian, Tanggal mulai perjanjian, Tanggal akhir perjanjian, Tanggal penerimaan perjanjian, Harga grosir, Margin, Jumlah pembelian

Email	Peristiwa	Penerima	Judul	Bidang
Marketplace Saluran.	Mitra AWS Marketplace Saluran.		ce Saluran.	
Perjanjian Dimulai — penawaran pribadi	AWS Marketplace Perjanjian untuk kontrak atau produk berlangganan telah dimulai dari penawaran pribadi dengan tanggal mulai future.	ISV	AWS Marketplace Kesepakatan telah dimulai	ID AWS akun pelanggan, Nama Produk, ID Produk, Nama Penawaran, ID Penawaran, ID Perjanjian, Tanggal mulai Perjanjian, Tanggal akhir Perjanjian, Jumlah pembelian

Email	Peristiwa	Penerima	Judul	Bidang
Perjanjian Dimulai — Penawara pribadi Mitra AWS Marketplace Saluran	AWS Marketplace Perjanjian untuk kontrak atau produk langganar telah dimulai dari penawara pribadi Mitra AWS Marketplace Saluran.	ISV	AWS Marketplace Kesepakatan telah dimulai.	ID AWS akun pelanggan, nama mitra saluran, ID AWS akun mitra saluran, nama produk, ID produk, nama penawaran, ID penawaran, ID perjanjian, tanggal mulai perjanjian, tanggal akhir perjanjian, harga grosir

Email	Peristiwa	Penerima	Judul	Bidang
Perjanjian Dimulai — Penawara pribadi	Penerimaan penawaran pribadi Mitra AWS Marketplace Saluran memulai perjanjian untuk kontrak atau produk subscription.	AWS Marketplace Mitra Saluran	AWS Marketplace Kesepakatan telah dimulai.	ID AWS akun pelanggan, nama ISV, ID AWS akun ISV, Nama Produk, ID Produk, Nama Penawaran, ID Penawaran, ID Perjanjian, Tanggal mulai Perjanjian, Tanggal akhir Perjanjian, Margin, Jumlah pembelian
Perjanjian Dibatalkan — Penawara pribadi atau publik	Pembatalan perjanjian penawaran pribadi atau publik.	ISV	Sebuah AWS Marketplace perjanjian dibatalkan.	ID AWS Akun Pelanggan, Nama Produk, ID Produk, Nama Penawaran, ID Penawaran, ID Perjanjian, Tanggal mulai Perjanjian, Tanggal akhir Perjanjian

Email	Peristiwa	Penerima	Judul	Bidang
Perjanjian Dibatalkan — Penawara pribadi Mitra AWS Marketplace Saluran	Pembatalan perjanjian penawara pribadi Mitra AWS Marketplace Saluran.	ISV	Sebuah AWS Marketplace perjanjian dibatalkan.	ID AWS akun pelanggan, Mitra saluran, ID AWS akun mitra Saluran, Nama Produk, ID Produk, Nama Penawaran, ID Penawaran, ID Perjanjian, Tanggal mulai perjanjian, Tanggal akhir perjanjian
Perjanjian Dibatalkan — Penawara Pribadi Mitra Saluran	Pembatalan perjanjian penawara pribadi Mitra AWS Marketplace Saluran.	AWS Marketplace Mitra Saluran	Sebuah AWS Marketplace perjanjian dibatalkan.	ID AWS akun pelanggan, ISV, ID AWS akun ISV, Nama Produk, ID Produk, Nama Penawaran, ID Penawaran, ID Perjanjian, Tanggal mulai Perjanjian, Tanggal akhir perjanjian
Kegagalan Kepatuhan — Penjual Catatan	AWS Marketplace Perjanjian gagal karena kegagalan pembayaran pelanggan.	ISV atau AWS Marketplace Channel Partner yang membuat penawara	Tindakan yang diperlukan: Kegagalan Pembuatan AWS Marketplace Perjanjian	ID AWS akun berlangganan, Nama Produk, ID Produk, Nama Penawaran, ID Penawaran, ID Perjanjian

Email	Peristiwa	Penerima	Judul	Bidang
Kegagalan Pembayar — Penjual Catatan	AWS Marketplace Perjanjian gagal karena kegagalan pembayar n pelanggan	ISV atau AWS Marketplace Channel Partner yang membuat penawaran	Kegagalan pembayar n untuk suatu AWS Marketplace perjanjian	ID AWS Akun Pelanggan, Nama Produk, ID Produk, Nama Penawaran, ID Penawaran, ID Perjanjian
Perjanjian Diganti - Penawaran pribadi	AWS Marketplace Perjanjian digantikan oleh perjanjian lain.	ISV	AWS Marketplace Perjanjian telah diganti	Nama perusahaan, ID AWS akun Pelanggan, Nama Produk, ID Produk, Nama Penawaran, ID Penawaran, ID perjanjian baru, Tanggal mulai perjanjian, Tanggal akhir perjanjian, Jumlah pembelian baru
Perjanjian Diganti — Penawaran Pribadi Mitra Saluran	AWS Marketplace Perjanjian digantikan oleh perjanjian lain.	ISV	AWS Marketplace Perjanjian telah diganti	Nama perusahaan, ID AWS akun pelanggan, mitra saluran, ID AWS akun mitra saluran, nama produk, ID produk, nama penawaran, ID penawaran, ID perjanjian baru, tanggal mulai perjanjian, tanggal akhir perjanjian, harga grosir

Email	Peristiwa	Penerima	Judul	Bidang
Perjanjian Diganti — Penawaran Pribadi Mitra Saluran	AWS Marketplace Perjanjian digantikan oleh perjanjian lain.	AWS Marketplace Mitra Saluran	AWS Marketplace Perjanjian telah diganti	Nama perusahaan, ID AWS akun pelanggan, nama ISV, ID AWS akun ISV, Nama Produk, ID Produk, Nama Penawaran, ID Penawaran, ID perjanjian baru, Tanggal mulai perjanjian, Tanggal akhir perjanjian, Harga grosir, Margin, Jumlah pembelian baru

Pencairan

ISVs dan Channel Partners akan mendapatkan pemberitahuan email jika pencairan mereka dihentikan sementara karena rincian rekening bank yang tidak valid.

Email	Peristiwa	Penerima	Judul	Bidang
Pencairan Dijeda — Informasi Rekening Bank Tidak Valid	AWS Marketplace menjeda pencairan ke penjual karena detail rekening bank yang tidak valid.	ISV dan Mitra AWS Marketplace Saluran	Pencairan Dijeda	ARN Sumber Daya

Resource ARN kolom menunjukkan rekening bank Amazon Resource Number (ARN) yang tidak valid. Anda dapat memperbaiki masalah ini dengan [menambahkan rekening bank](#) di Portal AWS Marketplace Manajemen (AMMP).

Menambahkan detail rekening bank

Tambahkan detail rekening bank di Portal AWS Marketplace Manajemen (AMMP) untuk memperbaiki kesalahan ARN rekening bank yang tidak valid dalam pencairan.

Untuk menambahkan detail rekening bank

1. Masuk ke akun [Portal Manajemen AWS Marketplace](#) dengan AWS Marketplace penjual Anda.
2. Pada bilah menu, pilih Pengaturan.
3. Pilih tab Informasi pembayaran.
4. Di bagian Rekening bank, pilih Tambah rekening bank.
5. Tambahkan detail rekening bank dan tinjau keakuratannya. Untuk informasi lebih lanjut, lihat [Langkah 3: Berikan informasi rekening bank](#).
6. Untuk menerima pembayaran dari pembeli, di bawah metode Pencairan, pilih Tambahkan metode pencairan.
7. Jika diminta untuk memverifikasi Know Your Customer (KYC) dan rekening bank, ikuti langkah-langkah verifikasi.
8. Simpan perubahan Anda.

Deskripsi bidang

Tabel berikut menunjukkan deskripsi bidang sebagaimana dimaksud dalam [Penawaran](#) dan [Perjanjian](#) tabel.

Bidang	Deskripsi			
Nama perusahaan pelanggan	Nama perusahaan pelanggan.			
ID AWS Rekening Nasabah	ID AWS akun berlangganan produk.			

Bidang	Deskripsi			
Nama ISV	Nama bisnis penjual.			
ID AWS akun ISV	ID AWS akun penjual.			
Nama produk	Judul produk.			
ID Produk	Ramah, pengidentifikasi unik untuk produk perangkat lunak.			
Nama penawaran	Judul penawaran .			
ID Penawaran	Pengidentifikasi penawaran yang ditandatangani pembeli.			
Menawarkan visibilitas	Apakah penawaran itu adalah penawaran kontrak publik, swasta, atau perusahaan.			

Bidang	Deskripsi			
ID Perjanjian	Referensi umpan data perjanjian unik untuk perjanjian yang ditandatangani antara pengusul dan penerima untuk mulai menggunakan produk.			
Tanggal mulai perjanjian	Tanggal langganan produk pelanggan dimulai, dalam formatMM-DD-YYYY . Tanggal ini bisa berbeda dari tanggal penerimaan untuk perjanjian bertanggal masa depan.			
Tanggal penerimaan perjanjian	Tanggal ketika pelanggan berlangganan produk, dalam formatMM-DD-YYYY .			

Bidang	Deskripsi			
Tanggal akhir perjanjian	Tanggal ketika kontrak berakhir, diformat dalam format. MM-DD-YYYY Untuk meteran atau pay-as-you-go langganan, tanggal ini diatur ke. JAN-1-9999			
Tanggal akhir perjanjian	Tanggal ketika kontrak berakhir, diformat dalam format. MM-DD-YYYY Untuk meteran atau pay-as-you-go langganan, tanggal ini diatur ke. JAN-1-9999			

Bidang	Deskripsi			
Jumlah pembelian	Estimasi biaya perjanjian, atau dikenal sebagai nilai kontrak total. Ini berlaku untuk SaaS, layanan profesional, dan jenis produk server dan jenis kontrak atau harga tahunan.			
Nama perusahaan mitra saluran	Nama akun yang membeli produk atau layanan dengan biaya grosir dari ISV untuk dijual kembali ke pelanggan.			
ID AWS akun mitra saluran	ID AWS akun Mitra AWS Marketplace Saluran yang membeli produk atau layanan dari ISV untuk dijual kembali kepada pelanggan.			

Bidang	Deskripsi			
Harga grosir	Biaya grosir dari ISV untuk menjual kembali produk ke Mitra AWS Marketplace Saluran.			
Kode mata uang	Penawaran harga mata uang terkait dengan perkiraan biaya perjanjian.			
ID perjanjian baru	ID Perjanjian perjanjian yang diperbarui atau ditingkatkan.			
Penawaran tanggal diterbitkan	Tanggal ketika penjual menerbitkan penawaran.			
Tanggal kedaluwarsa penawaran	Tanggal ketika penawaran berakhir.			
ID Peluang	Pengidentifikasi unik untuk peluang terdaftar .			

Bidang	Deskripsi			
Durasi otorisasi penjualan	Lamanya waktu pengecer berwenang untuk membuat penawaran menggunakan diskon, sebagaimana ditentukan dalam otorisasi penjualan.			

Kelola notifikasi

Topik berikut menjelaskan cara mengelola notifikasi email untuk acara.

Menambahkan atau memperbarui alamat email

Anda dapat menambahkan hingga 10 alamat email untuk notifikasi email khusus menggunakan Portal Manajemen AWS Marketplace.

Untuk menambah atau memperbarui alamat email

1. Masuk ke [Portal Manajemen AWS Marketplace](#).
2. Dari Pengaturan, pilih tab Pemberitahuan.
3. Di bawah Email untuk pemberitahuan kustom, pilih Tambahkan alamat email.
4. Untuk detail Penerima, masukkan alamat email khusus di bidang Alamat email.
5. (Opsional) Pilih Tambahkan penerima baru untuk menambahkan alamat email lain (total hingga 10).
6. Pilih Kirim.

Berhenti berlangganan penerima dari notifikasi

Anda dapat menghapus alamat email sehingga penerima berhenti berlangganan dari notifikasi email khusus.

Untuk berhenti berlangganan penerima dari pemberitahuan acara

1. Masuk ke [Portal Manajemen AWS Marketplace](#).
2. Dari Pengaturan pilih tab Pemberitahuan.
3. Di bawah Email untuk pemberitahuan kustom, pilih Perbarui alamat email.
4. Untuk detail Penerima, pilih Hapus untuk menghapus alamat email.
5. Pilih Kirim.

Note

Anda juga dapat berhenti berlangganan menggunakan tautan di email.

EventBridge Acara Amazon

Sebagai penjual, Anda dapat menggunakan Amazon EventBridge untuk menerima pemberitahuan untuk acara di AWS Marketplace. Misalnya, Anda dapat menerima acara dari AWS Marketplace saat penawaran dibuat. Acara ini berisi detail seperti ID, tanggal kedaluwarsa, dan detail produk. EventBridge adalah layanan bus acara yang dapat Anda gunakan untuk menghubungkan aplikasi Anda dengan data dari berbagai sumber. Untuk informasi selengkapnya, lihat [Panduan EventBridge Pengguna Amazon](#). Bagian berikut memberikan informasi terperinci tentang peristiwa di bawah layanan Katalog Marketplace di EventBridge konsol.

Topik

- [Acara untuk penawaran baru](#)
- [Acara untuk set perubahan](#)
- [Acara untuk laporan ringkasan keamanan](#)
- [Acara untuk pencairan](#)

Topik ini

Tindakan oleh penjual	Acara diterima	Topik terkait
Vendor perangkat lunak independen (ISV) membuat penawaran dan membuatnya tersedia untuk pembelian	Offer Released	the section called “Acara untuk penawaran baru”
Produk ISV digunakan oleh mitra saluran untuk membuat penawaran	Offer Released	the section called “Acara untuk penawaran baru”
Mitra saluran membuat penawaran	Offer Released	the section called “Acara untuk penawaran baru”
Ubah set berhasil	Change Set Succeeded	the section called “Acara untuk set perubahan”
Gagal mengubah set	Change Set Failed	the section called “Acara untuk set perubahan”
Set perubahan dibatalkan	Change Set Cancelled	the section called “Acara untuk set perubahan”
Kerentanan keamanan terdeteksi pada produk ISV	Products Security Report Created	the section called “Acara untuk laporan ringkasan keamanan”

Acara untuk penawaran baru

Saat penjual membuat penawaran dan membuatnya tersedia untuk dibeli, mereka dapat menerima acara dengan jenis detail berikut: Offer Released.

Note

Untuk informasi tentang cara membuat EventBridge aturan, lihat [EventBridge Aturan Amazon](#) di Panduan EventBridge Pengguna Amazon.

Berikut ini adalah contoh badan acara untuk penawaran baru yang dibuat oleh ISV.

```

{
  "version": "0",
  "id": "01234567-0123-0123-0123-0123456789ab",
  "detail-type": "Offer Released",
  "source": "aws.marketplacecatalog",
  "account": "123456789012",
  "time": "2023-08-26T00:00:00Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/Offer/offer-1234567890123"
  ],
  "detail": {
    "requestId": "3d4c9f9b-b809-4f5e-9fac-a9ae98b05cbb",
    "catalog": "AWSMarketplace",
    "offer": {
      "id": "offer-1234567890123",
      "arn": "arn:aws:catalog:us-east-1:123456789012:Offer/offer-1234567890123",
      "name": "Offer Name",
      "expirationDate": "2025-08-26T00:00:00Z"
    },
    "product": {
      "id": "bbbbaaaa-abcd-1111-abcd-666666666666",
      "arn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/SaaSProduct/bbbbbaaaa-abcd-1111-abcd-666666666666",
      "title": "Product Title"
    },
    "manufacturer": {
      "accountId": "123456789012",
      "name": "Manufacturer Account Name"
    },
    "sellerOfRecord": {
      "accountId": "123456789012",
      "name": "Seller Account Name"
    },
    "targetedBuyerAccountIds": [
      "999988887777",
      "111122223333"
    ]
  }
}

```

Berikut ini adalah contoh badan acara ketika produk ISV digunakan oleh mitra saluran untuk membuat penawaran.

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-0123456789ab",
  "detail-type": "Offer Released",
  "source": "aws.marketplacecatalog",
  "account": "123456789012",
  "time": "2023-08-26T00:00:00Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:aws-marketplace:us-east-1:987654321098:AWSMarketplace/Offer/offer-1234567890123"
  ],
  "detail": {
    "requestId": "3d4c9f9b-b809-4f5e-9fac-a9ae98b05cbb",
    "catalog": "AWSMarketplace",
    "offer": {
      "id": "offer-1234567890123",
      "arn": "arn:aws:catalog:us-east-1:987654321098:Offer/offer-1234567890123",
      "name": "Offer Name",
      "expirationDate": "2025-08-26T00:00:00Z"
    },
    "product": {
      "id": "bbbbaaaa-abcd-1111-abcd-666666666666",
      "arn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/SaaSProduct/bbbbbaaaa-abcd-1111-abcd-666666666666",
      "title": "Product Title"
    },
    "manufacturer": {
      "accountId": "123456789012",
      "name": "Manufacturer Account Name"
    },
    "sellerOfRecord": {
      "accountId": "987654321098",
      "name": "Seller Account Name"
    },
    "targetedBuyerAccountIds": ["999988887777", "111122223333"],
  }
}
```

Berikut ini adalah contoh badan acara ketika mitra saluran membuat penawaran.

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-0123456789ab",
  "detail-type": "Offer Released",
  "source": "aws.marketplacecatalog",
  "account": "987654321098",
  "time": "2023-08-26T00:00:00Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:aws-marketplace:us-east-1:987654321098:AWSMarketplace/Offer/offer-1234567890123"
  ],
  "detail": {
    "requestId": "3d4c9f9b-b809-4f5e-9fac-a9ae98b05cbb",
    "catalog": "AWSMarketplace",
    "offer": {
      "id": "offer-1234567890123",
      "arn": "arn:aws:catalog:us-east-1:987654321098:Offer/offer-1234567890123",
      "name": "Offer Name",
      "expirationDate": "2025-08-26T00:00:00Z"
    },
    "product": {
      "id": "bbbbaaaa-abcd-1111-abcd-666666666666",
      "arn": "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/SaaSProduct/bbbbbaaaa-abcd-1111-abcd-666666666666",
      "title": "Product Title"
    },
    "manufacturer": {
      "accountId": "123456789012",
      "name": "Manufacturer Account Name"
    },
    "sellerOfRecord": {
      "accountId": "987654321098",
      "name": "Seller Account Name"
    },
    "targetedBuyerAccountIds": ["999988887777", "111122223333"],
  }
}
```

Acara untuk set perubahan

Ketika set perubahan selesai, penjual, mitra saluran, dan administrator pasar pribadi dapat menerima acara. API AWS Marketplace Katalog mengirimkan peristiwa ketika set perubahan selesai dengan status berhasil, gagal, atau dibatalkan. Sumber untuk peristiwa ini adalah `aws.marketplacecatalog`, dan nilai tipe detail yang mungkin adalah `Change Set Succeeded`, `Change Set Failed`, dan `Change Set Cancelled`.

Note

Untuk informasi tentang set perubahan, lihat [Bekerja dengan set perubahan](#) di Referensi API AWS Marketplace Katalog.

Setiap peristiwa berisi detail permintaan perubahan, seperti ID set perubahan, nama set perubahan, jenis detail peristiwa, kode kegagalan (untuk permintaan gagal), dan waktu mulai dan akhir permintaan. Hal ini memungkinkan Anda untuk memantau set perubahan Anda tanpa terus-menerus menanyakan `DescribeChangeSet` tindakan atau memeriksa status permintaan perubahan Anda. Portal Manajemen AWS Marketplace

Note

Untuk informasi tentang cara membuat EventBridge aturan, lihat [EventBridge Aturan Amazon](#) di Panduan EventBridge Pengguna Amazon.

Berikut ini adalah contoh badan acara untuk jenis `Change Set Succeeded` detail.

```
{
  "version": "0",
  "id": "01234567-0123-0123-0123-0123456789ab",
  "detail-type": "Change Set Succeeded",
  "source": "aws.marketplacecatalog",
  "account": "123456789012",
  "time": "2022-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/ChangeSet/76yesvf8y165pa4f98td2crtg"
```

```

],
"detail": {
  "requestId" : "3d4c9f9b-b809-4f5e-9fac-a9ae98b05cbb",
  "Catalog": "AWSMarketplace",
  "ChangeSetId": "76yesvf8y165pa4f98td2crtg",
  "ChangeSetName": "Create my product",
  "StartTime": "2018-02-27T13:45:22Z",
  "EndTime": "2018-02-27T14:55:22Z"
}
}

```

Berikut ini adalah contoh badan acara untuk jenis Change Set Failed detail.

```

{
  "version": "0",
  "id": "01234567-0123-0123-0123-0123456789ab",
  "detail-type": "Change Set Failed",
  "source": "aws.marketplacecatalog",
  "account": "123456789012",
  "time": "2022-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": [
    "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/76yesvf8y165pa4f98td2crtg"
  ],
  "detail": {
    "requestId" : "3d4c9f9b-b809-4f5e-9fac-a9ae98b05cbb",
    "Catalog": "AWSMarketplace",
    "ChangeSetId": "76yesvf8y165pa4f98td2crtg",
    "ChangeSetName": "Create my product",
    "StartTime": "2018-02-27T13:45:22Z",
    "EndTime": "2018-02-27T14:55:22Z",
    "FailureCode": "CLIENT_ERROR"
  }
}

```

Berikut ini adalah contoh badan acara untuk jenis Change Set Cancelled detail.

```

{
  "version": "0",
  "id": "01234567-0123-0123-0123-0123456789ab",
  "detail-type": "Change Set Cancelled",
  "source": "aws.marketplacecatalog",

```

```

"account": "123456789012",
"time": "2022-11-01T13:12:22Z",
"region": "us-east-1",
"resources": [
  "arn:aws:aws-marketplace:us-east-1:123456789012:AWSMarketplace/
ChangeSet/76yesvf8y165pa4f98td2crtg"
],
"detail": {
  "requestId" : "3d4c9f9b-b809-4f5e-9fac-a9ae98b05cbb",
  "Catalog": "AWSMarketplace",
  "ChangeSetId": "76yesvf8y165pa4f98td2crtg",
  "ChangeSetName": "Create my product",
  "StartTime": "2018-02-27T13:45:22Z",
  "EndTime": "2018-02-27T14:55:22Z"
}
}

```

Acara untuk laporan ringkasan keamanan

Ketika kerentanan keamanan terdeteksi pada produk penjual, mereka dapat menerima acara laporan ringkasan dan pengingat berkala untuk masalah produk yang luar biasa. Sumber untuk acara ini adalah `aws.marketplacecatalog`, dan jenis detailnya adalah `Products Security Report Created`.

Setiap acara mencakup ringkasan jumlah produk dan versi dengan masalah yang terdeteksi, hitungan berapa banyak versi terbaru yang terpengaruh, dan tanggal ketika resolusi diperlukan untuk mencegah pembatasan sementara produk atau versi ini.

Note

Untuk informasi tentang cara membuat EventBridge aturan, lihat [EventBridge Aturan Amazon](#) di Panduan EventBridge Pengguna Amazon.

Untuk detail tentang mengelola peristiwa keamanan, lihat [Cara meningkatkan keamanan katalog produk Anda di](#) posting AWS Marketplace blog di AWS Blog.

Berikut ini adalah contoh badan acara untuk jenis `Products Security Report Created` detail.

```

{
  "version": "0",

```

```

"id": "01234567-0123-0123-0123-0123456789ab",
"detail-type": "Products Security Report Created",
"source": "aws.marketplacecatalog",
"account": "123456789012",
"time": "2023-10-31T00:00:00Z",
"region": "us-east-1",
"resources": [],
"detail": {
  "numberOfProductsWithIssues": 1,
  "numberOfVersionsWithIssues": 1,
  "numberOfLatestVersionsWithIssues": 1,
  "newIssuesFound": true,
  "upcomingResolutionDueDate": "2023-12-01T00:00:00Z",
  "requestId": "533fa17d-3e97-5051-bcaf-1fae45fb3f8b"
}
}

```

Acara untuk pencairan

Ketika pencairan ke rekening bank penjual gagal karena rincian rekening bank yang tidak valid, AWS Marketplace ISVs dan mitra saluran dapat menerima acara.

Dalam kode acara JSON berikut, `source` nilai untuk peristiwa ini adalah `aws.marketplace`, dan `detail-type` nilainya adalah `Disbursement Paused`. `resources` Nilai menunjukkan rekening bank yang tidak valid Amazon Resource Number (ARN).

```

{
  "version": "0",
  "id": "01234567-0123-0123-0123-0123456789ab",
  "detail-type": "Disbursement Paused",
  "source": "aws.marketplace",
  "account": "<account id of end user>",
  "time": "2022-11-01T13:12:22Z",
  "region": "us-east-1",
  "resources": ["arn:aws:payments:us-east-1:1234567890:paymentinstrument:123"],
  "detail": {
    "requestId": "3d4c9f9b-b809-4f5e-9fac-a9ae98b05cbb",
    "catalog": "AWSMarketplace"
  }
}

```

Untuk mengatasi masalah detail rekening bank yang tidak valid, tambahkan detail rekening bank Anda di Portal AWS Marketplace Manajemen. Untuk petunjuk, lihat [Untuk menambahkan detail rekening bank](#).

Untuk informasi selengkapnya tentang membuat EventBridge aturan Amazon, lihat [Aturan EventBridge di Amazon](#) di Panduan EventBridge Pengguna Amazon.

Laporan penjual, umpan data, dan dasbor di AWS Marketplace

Sebagai AWS Marketplace penjual, Anda dapat menggunakan alat berikut untuk mengumpulkan dan menganalisis informasi tentang penjualan produk Anda:

- [Laporan](#) yang AWS Marketplace dihasilkan secara otomatis dan tersedia untuk semua AWS Marketplace penjual terdaftar.
- [API](#) yang dapat Anda gunakan untuk mengakses bagian dari laporan tersebut.
- [Umpan data](#) yang memberikan informasi pelanggan tambahan yang dapat Anda gunakan untuk mengidentifikasi informasi pelanggan untuk transaksi yang tercantum dalam laporan.
- [Dasbor](#) yang didukung oleh [Amazon QuickSight](#) dengan bagan, grafik, dan wawasan yang membantu Anda mengakses dan menganalisis data keuangan.

AWS Marketplace menyediakan data sebanyak mungkin dalam laporan, umpan data, dan dasbor sambil mengikuti hal-hal berikut:

- Standar dan prinsip Amazon untuk melindungi data pelanggan.
- Syarat dan ketentuan yang diterima pembeli ketika mereka membeli produk AWS Marketplace. Sebagai penjual, Anda terikat kontrak untuk mengelola data pembeli dengan aman dan menghapus data atas permintaan pembeli.

Note

Untuk membatalkan email pemberitahuan untuk laporan, hubungi tim [Operasi AWS Marketplace Penjual](#).

Untuk informasi selengkapnya tentang laporan, data, dan dasbor di AWS Marketplace, lihat topik berikut.

Topik

- [Umpan data pengiriman penjual di AWS Marketplace](#)
- [Laporan penjual di AWS Marketplace](#)

- [Laporan tambahan](#)
- [Dasbor penjual](#)

Umpan data pengiriman penjual di AWS Marketplace

AWS Marketplace menyediakan umpan data sebagai mekanisme untuk mengirim informasi terstruktur, up-to-date produk, dan pelanggan dari AWS Marketplace sistem ke bucket Amazon S3 penjual untuk ETL (ekstrak, transformasi, dan muat) antara alat intelijen bisnis milik penjual. Umpan data mengumpulkan dan mengirimkan file nilai dipisahkan koma (CSV) ke bucket Amazon S3 terenkripsi yang Anda berikan. Umpan data dihasilkan dalam sehari, dan berisi 24 jam data dari hari sebelumnya. Bagian berikut memberikan ikhtisar umpan data dan menjelaskan cara mengakses dan menggunakannya. Bagian selanjutnya menjelaskan setiap umpan data.

Data transaksional dikirim dan ditambahkan dalam struktur bi-temporal sehingga penjual dapat menyimpan dan menanyakan data di sepanjang dua garis waktu dengan stempel waktu untuk keduanya

- waktu yang valid: ketika sebuah fakta terjadi di dunia nyata (“apa yang Anda ketahui”)
- waktu sistem: ketika fakta itu dicatat ke database (“ketika Anda mengetahuinya”).

Umpan data dikirim setiap hari pada tengah malam UTC setelah pembaruan dari hari sebelumnya yang berisi 24 jam data dari hari sebelumnya. Pembaruan dapat ditentukan oleh pelanggan yang berlangganan, pelanggan yang ditagih, atau AWS mencairkan pembayaran.

Topik

- [Penyimpanan dan struktur umpan AWS Marketplace data](#)
- [Mengakses umpan data](#)
- [Mengumpulkan dan menganalisis data dengan umpan data](#)
- [Ikhtisar tabel umpan data](#)
- [Contoh kueri umpan data](#)
- [Umpan data](#)

Penyimpanan dan struktur umpan AWS Marketplace data

AWS Marketplace menyediakan umpan data sebagai mekanisme untuk mengirim informasi terstruktur, up-to-date produk, dan pelanggan dari AWS Marketplace sistem ke bucket Amazon S3 penjual untuk ETL (ekstrak, transformasi, dan muat) antara alat intelijen bisnis milik penjual. Topik ini memberikan informasi lebih lanjut tentang struktur dan penyimpanan umpan data.

Umpan data mengumpulkan dan mengirimkan file nilai dipisahkan koma (CSV) ke bucket Amazon S3 terenkripsi yang Anda berikan. File CSV memiliki karakteristik sebagai berikut:

- Mereka mengikuti [standar 4180](#).
- Pengkodean karakter adalah UTF-8 tanpa BOM.
- Koma digunakan sebagai pemisah antar nilai.
- Bidang diloloskan oleh tanda kutip ganda.
- \n adalah karakter umpan baris.
- Tanggal dilaporkan dalam zona waktu UTC, dalam format tanggal dan waktu ISO 8601, dan akurat dalam 1 detik.
- Semua *_period_start_date dan *_period_end_date nilai bersifat inklusif, yang berarti itu 23:59:59 adalah stempel waktu terakhir yang mungkin untuk setiap hari.
- Semua bidang moneter didahului dengan bidang mata uang.
- Bidang moneter menggunakan karakter period (.) sebagai pemisah desimal, dan jangan gunakan koma (,) sebagai pemisah ribuan.

Umpan data dihasilkan dan disimpan sebagai berikut:

- Umpan data dihasilkan dalam sehari, dan berisi 24 jam data dari hari sebelumnya.
- Di bucket Amazon S3, umpan data diatur berdasarkan bulan menggunakan format berikut:

bucket-name/data-feed-name_version/year=YYYY/month=MM/data.csv

- Karena setiap umpan data harian dihasilkan, itu ditambahkan ke file CSV yang ada untuk bulan itu. Saat bulan baru dimulai, file CSV baru dibuat untuk setiap umpan data.
- [Informasi dalam umpan data diisi kembali dari 2010/01/01 hingga 2020/04/30 \(inklusif\) dan tersedia dalam file CSV di subfolder.](#) year=2010/month=01

Anda mungkin melihat kasus di mana file bulan berjalan untuk umpan data tertentu hanya berisi header kolom, dan tidak ada data. Ini berarti bahwa tidak ada entri baru untuk bulan itu untuk

umpan. Ini dapat terjadi dengan umpan data yang diperbarui lebih jarang, seperti umpan produk. Dalam kasus ini, data tersedia di folder yang diisi ulang.

- Di Amazon S3, Anda dapat membuat kebijakan [siklus hidup Amazon S3](#) untuk mengelola berapa lama menyimpan file di bucket.
- Anda dapat mengonfigurasi Amazon SNS untuk memberi tahu Anda saat data dikirimkan ke bucket Amazon S3 terenkripsi. Untuk informasi tentang cara mengonfigurasi notifikasi, lihat [Memulai Amazon SNS](#) di Panduan Pengembang Layanan Pemberitahuan Sederhana Amazon.

Historisasi data

Setiap umpan data menyertakan kolom yang mendokumentasikan riwayat data. Kecuali untuk `valid_to`, kolom ini umum untuk semua umpan data. Mereka disertakan sebagai skema riwayat umum dan berguna dalam menanyakan data.

Nama kolom	Deskripsi
<code>valid_dari</code>	Tanggal pertama dimana nilai untuk kunci utama valid dalam kaitannya dengan nilai untuk bidang lain.
<code>valid_to</code>	Kolom ini hanya ditampilkan pada umpan data Alamat dan selalu kosong.
<code>insert_date</code>	Tanggal catatan dimasukkan ke dalam umpan data.
<code>update_date</code>	Tanggal rekor terakhir diperbarui.
<code>delete_date</code>	Kolom ini selalu kosong.

Berikut ini menunjukkan contoh kolom ini.

<code>valid_dari</code>	<code>valid_to</code>	<code>insert_date</code>	<code>update_date</code>	<code>delete_date</code>
2018-12-12T 02:00:00 Z		2018-12-12T 02:00:00 Z	2018-12-12T 02:00:00 Z	

valid_dari	valid_to	insert_date	update_date	delete_date
2019-03-29T 03:00:00 Z		2019-03-29T 03:00:00 Z	2019-03-29T 03:00:00 Z	
2019-03-29T 03:00:00 Z		2019-03-29T 03:00:00 Z	2019-04-28T 03:00:00 Z	

update_date Bidang valid_from dan bersama-sama membentuk model data bi-temporal. valid_from Bidang, seperti namanya, memberi tahu Anda kapan item tersebut valid dari. Jika item diedit, itu dapat memiliki beberapa catatan dalam umpan, masing-masing dengan valid_from tanggal yang berbeda update_date, tetapi sama. Misalnya, untuk menemukan nilai saat ini untuk suatu item, Anda akan menemukan catatan dengan yang terbaru update_date, dari daftar catatan dengan valid_from tanggal terbaru.

Pada contoh di atas, catatan awalnya dibuat 2018-12-12. Itu kemudian diubah pada 2019-03-29 (misalnya, jika alamat dalam catatan berubah). Kemudian, pada 2019-04-28, perubahan alamat diperbaiki (jadi valid_from tidak berubah, tetapi berubah). update_date Memperbaiki alamat (peristiwa langka) secara surut mengubah catatan dari valid_from tanggal asli, sehingga bidang itu tidak berubah. Kueri untuk menemukan yang terbaru valid_from akan mengembalikan dua catatan, satu dengan yang terbaru update_date memberi Anda catatan saat ini yang sebenarnya.

Mengakses umpan data

Dengan AWS Marketplace, Anda dapat menggunakan umpan data sebagai mekanisme untuk mengirim informasi terstruktur, up-to-date produk, dan pelanggan dari AWS Marketplace sistem bucket Amazon S3 untuk ETL (ekstrak, transformasi, dan muat) antara alat intelijen bisnis milik penjual. Anda perlu mengonfigurasi lingkungan Anda untuk menerima umpan data ke bucket Amazon S3 terenkripsi. Topik ini menunjukkan kepada Anda cara mengakses dan berhenti berlangganan dari umpan data.

Topik

- [Mengakses umpan data](#)
- [Kebijakan umpan data](#)
- [Berhenti berlangganan dari umpan data](#)

Mengakses umpan data

1. Alokasikan intelijen bisnis atau insinyur data dengan pengalaman SQL dan ETL (ekstrak, transformasi, muat). Orang ini juga membutuhkan pengalaman mengatur APIs.
2. Siapkan bucket Amazon Simple Storage Service dan berlangganan feed data. Gunakan ID akun AWS penjual yang terkait dengan daftar produk Marketplace Anda. Untuk melakukannya, Anda dapat [menonton YouTube video ini](#) atau mengikuti langkah-langkah di bawah ini.

Video dan langkah-langkahnya menjelaskan cara menggunakan [AWS CloudFormation templat](#) yang membantu menyederhanakan konfigurasi.

- a. Buka browser web dan masuk ke [Portal Manajemen AWS Marketplace](#), lalu pergi ke [Mengatur penyimpanan data pelanggan](#).
- b. Pilih Buat sumber daya dengan AWS CloudFormation templat untuk membuka templat di AWS CloudFormation konsol di jendela lain.
- c. Dalam template, tentukan yang berikut dan kemudian pilih Berikutnya:
 - Nama tumpukan — Kumpulan sumber daya yang Anda buat untuk mengaktifkan akses ke umpan data.
 - Nama bucket Amazon S3 — Bucket untuk menyimpan umpan data.
 - (Opsional) Nama topik Amazon SNS — Topik untuk menerima pemberitahuan saat Amazon Simple Storage Service bucket.
- d. Pada halaman Tinjauan, konfirmasi entri Anda dan pilih Buat tumpukan. Ini akan membuka halaman baru dengan CloudFormation status dan detailnya.
- e. Dari tab Resources, salin Amazon Resource Names (ARNs) untuk sumber daya berikut dari CloudFormation halaman ke bidang di halaman AWS Marketplace [Menyiapkan penyimpanan data pelanggan](#):
 - Bucket Amazon S3 untuk menyimpan umpan data
 - AWS KMS kunci untuk mengenkripsi bucket Amazon S3
 - (Opsional) Topik Amazon SNS untuk menerima notifikasi saat AWS mengirimkan data baru ke bucket Amazon S3
- f. Pada halaman Mengatur penyimpanan data pelanggan, pilih Kirim.
- g. (Opsional) Edit kebijakan yang dibuat oleh CloudFormation template. Lihat [Kebijakan umpan data](#) untuk detail selengkapnya.

Anda sekarang berlangganan umpan data. Lain kali umpan data dihasilkan, Anda dapat mengakses data.

- Gunakan operasi ETL (ekstrak, transformasi, muat) untuk menghubungkan umpan data ke gudang data atau database relasional Anda.

 Note

Alat data memiliki kemampuan yang berbeda. Anda harus melibatkan insinyur intelijen bisnis atau insinyur data untuk mengatur integrasi agar sesuai dengan kemampuan alat Anda.

- Untuk menjalankan atau membuat kueri SQL, konfigurasi umpan data untuk menerapkan kunci primer dan asing di alat data Anda. Setiap umpan data mewakili tabel unik, dan Anda harus menyiapkan semua umpan data dalam skema data dengan hubungan entitas. Untuk informasi selengkapnya tentang tabel dan hubungan entitas, lihat [Ikhtisar tabel umpan data](#) di panduan ini.
- Siapkan Amazon Simple Notification Service untuk menyegarkan gudang data atau database relasional Anda secara otomatis. Anda dapat mengonfigurasi notifikasi Amazon SNS untuk mengirim peringatan saat data dari setiap umpan unik dikirimkan ke bucket Amazon S3. Pemberitahuan ini dapat dimanfaatkan untuk menyegarkan gudang data penjual secara otomatis ketika data baru diterima melalui umpan data, jika alat data penjual mendukung kemampuan ini. Lihat [Memulai Amazon SNS](#) di Panduan Pengembang Layanan Pemberitahuan Sederhana Amazon.

Contoh pemberitahuan:

```
{
  "mainExecutionId": "1bc08b11-ab4b-47e1-866a-9c8f38423a98",
  "executionId": "52e862a9-42d2-41e0-8010-810af84d39b1",
  "subscriptionId": "27ae3961-b13a-44bc-a1a7-365b2dc181fd",
  "processedFiles": [],
  "executionStatus": "SKIPPED",
  "errors": [],
  "feedType": "[data feed name]"
}
```

Pemberitahuan dapat memiliki executionStatus status berikut:

- SKIPPED— Penjual tidak memiliki data baru untuk hari itu.
 - COMPLETED— Kami mengirimkan feed dengan data baru.
 - FAILED— Pengiriman pakan memiliki masalah.
6. Validasi penyiapan dengan menjalankan kueri SQL. [Anda dapat menggunakan contoh kueri dalam panduan ini, atau kueri pada GitHub, di https://github.com/aws-samples/aws-marketplace-api-samples/tree/main/seller-data-feeds/queries](https://github.com/aws-samples/aws-marketplace-api-samples/tree/main/seller-data-feeds/queries)

 Note

Contoh pertanyaan dalam panduan ini ditulis untuk AWS Athena. Anda mungkin perlu memodifikasi kueri untuk digunakan dengan alat Anda.

7. Tentukan di mana pengguna bisnis ingin mengkonsumsi data. Sebagai contoh, Anda dapat:
- Ekspor data.csv dari gudang data atau database SQL Anda.
 - Hubungkan data Anda ke alat visualisasi seperti PowerBI atau Tableau.
 - Petakan data ke CRM, ERP, atau alat keuangan Anda, seperti Salesforce, Infor, atau Netsuite.

Untuk informasi selengkapnya tentang AWS CloudFormation templat, lihat [Bekerja dengan AWS CloudFormation templat](#) di Panduan AWS CloudFormation Pengguna.

Kebijakan umpan data

Saat bucket Amazon S3 Anda dibuat oleh CloudFormation template, bucket Amazon S3 akan membuat kebijakan untuk akses yang dilampirkan ke bucket tersebut, AWS KMS kuncinya, dan topik Amazon SNS. Kebijakan ini memungkinkan layanan AWS Marketplace laporan untuk menulis ke bucket dan topik SNS Anda dengan informasi umpan data. Setiap kebijakan akan memiliki bagian seperti berikut (contoh ini dari bucket Amazon S3).

```
{
  "Sid": "AwsMarketplaceDataFeedsAccess",
  "Effect": "Allow",
  "Principal": {
    "Service": "reports.marketplace.amazonaws.com"
  },
  "Action": [
    "s3:ListBucket",
```

```

        "s3:GetObject",
        "s3:PutObject",
        "s3:GetEncryptionConfiguration",
        "s3:GetBucketAcl",
        "s3:PutObjectAcl"
    ],
    "Resource": [
        "arn:aws:s3:::amzn-s3-demo-bucket",
        "arn:aws:s3:::amzn-s3-demo-bucket/*"
    ]
},

```

Dalam kebijakan ini, AWS Marketplace gunakan prinsip `reports.marketplace.amazonaws.com` layanan untuk mendorong data ke bucket Amazon S3. Anda menentukan `amzn-s3-demo-bucket` dalam CloudFormation template.

Saat layanan AWS Marketplace laporan memanggil Amazon S3, AWS KMS, atau Amazon SNS, layanan ini menyediakan ARN dari data yang ingin ditulis ke bucket. Untuk memastikan bahwa satu-satunya data yang ditulis ke bucket Anda adalah data yang ditulis atas nama Anda, Anda dapat menentukan `aws:SourceArn` dalam kondisi kebijakan. Dalam contoh berikut, Anda harus mengganti `account-id` dengan ID untuk Anda Akun AWS.

```

{
  "Sid": "AwsMarketplaceDataFeedsAccess",
  "Effect": "Allow",
  "Principal": {
    "Service": "reports.marketplace.amazonaws.com"
  },
  "Action": [
    "s3:ListBucket",
    "s3:GetObject",
    "s3:PutObject",
    "s3:GetEncryptionConfiguration",
    "s3:GetBucketAcl",
    "s3:PutObjectAcl"
  ],
  "Resource": [
    "arn:aws:s3:::amzn-s3-demo-bucket",
    "arn:aws:s3:::amzn-s3-demo-bucket/*"
  ],
  "Condition": {

```

```
    "StringEquals": {
      "aws:SourceAccount": "account-id",
      "aws:SourceArn": ["arn:aws:marketplace::account-id:AWSMarketplace/SellerDataSubscription/DataFeeds_V1",
        "arn:aws:marketplace::account-id:AWSMarketplace/SellerDataSubscription/Example-Report"]
    }
  },
```

Berhenti berlangganan dari umpan data

Buka browser web dan masuk ke [Portal AWS Marketplace Manajemen](#). Kemudian, buka [halaman Hubungi kami](#) untuk mengirimkan permintaan berhenti berlangganan ke tim Operasi AWS Marketplace Penjual. Permintaan berhenti berlangganan dapat memakan waktu hingga 10 hari kerja untuk diproses.

Mengumpulkan dan menganalisis data dengan umpan data

AWS Marketplace menyediakan umpan data sebagai mekanisme untuk mengirim informasi terstruktur, up-to-date produk, dan pelanggan dari AWS Marketplace sistem ke bucket Amazon S3 penjual untuk ETL (ekstrak, transformasi, dan muat) antara alat intelijen bisnis milik penjual. Saat data tersedia di bucket Amazon S3, Anda dapat menggunakan umpan data dengan cara berikut:

- Unduh file.CSV dari bucket Amazon S3 yang Anda buat sehingga Anda dapat melihat data [Mengakses umpan data](#) dalam spreadsheet.
- Gunakan ETL (ekstrak, transformasi, dan muat), kueri SQL, alat analisis bisnis untuk mengumpulkan dan menganalisis data.

Anda dapat menggunakan AWS layanan untuk mengumpulkan dan menganalisis data, atau alat pihak ketiga apa pun yang dapat melakukan analisis kumpulan data berbasis CSV.

Untuk informasi selengkapnya tentang umpan data untuk mengumpulkan dan menganalisis data, lihat contoh berikut.

Contoh: Gunakan AWS layanan untuk mengumpulkan dan menganalisis data

Prosedur berikut mengasumsikan bahwa Anda telah mengonfigurasi lingkungan untuk menerima umpan data ke bucket Amazon S3 dan bucket berisi umpan data.

Mengumpulkan dan menganalisis data dari umpan data

1. Dari [AWS Glue konsol](#), [buat crawler](#) untuk terhubung ke bucket Amazon S3 yang menyimpan umpan data, mengekstrak data yang Anda inginkan, dan membuat tabel metadata di AWS Glue Data Catalog

Untuk informasi selengkapnya AWS Glue, lihat [Panduan AWS Glue Pengembang](#).

2. Dari [konsol Athena](#), [jalankan kueri SQL pada data di file AWS Glue Data Catalog](#)

Untuk informasi lebih lanjut tentang Athena, lihat Panduan Pengguna [Amazon Athena](#).

3. Dari [QuickSight konsol](#), [buat analisis](#) dan kemudian [buat visual](#) data.

Untuk informasi selengkapnya QuickSight, lihat [Panduan QuickSight Pengguna Amazon](#).

Untuk contoh terperinci tentang salah satu cara menggunakan AWS layanan untuk mengumpulkan dan menganalisis data dalam umpan data, lihat [Menggunakan Layanan Pengiriman Umpan Data Penjual, Amazon Athena, QuickSight dan untuk membuat laporan penjual](#) di AWS Marketplace Blog.

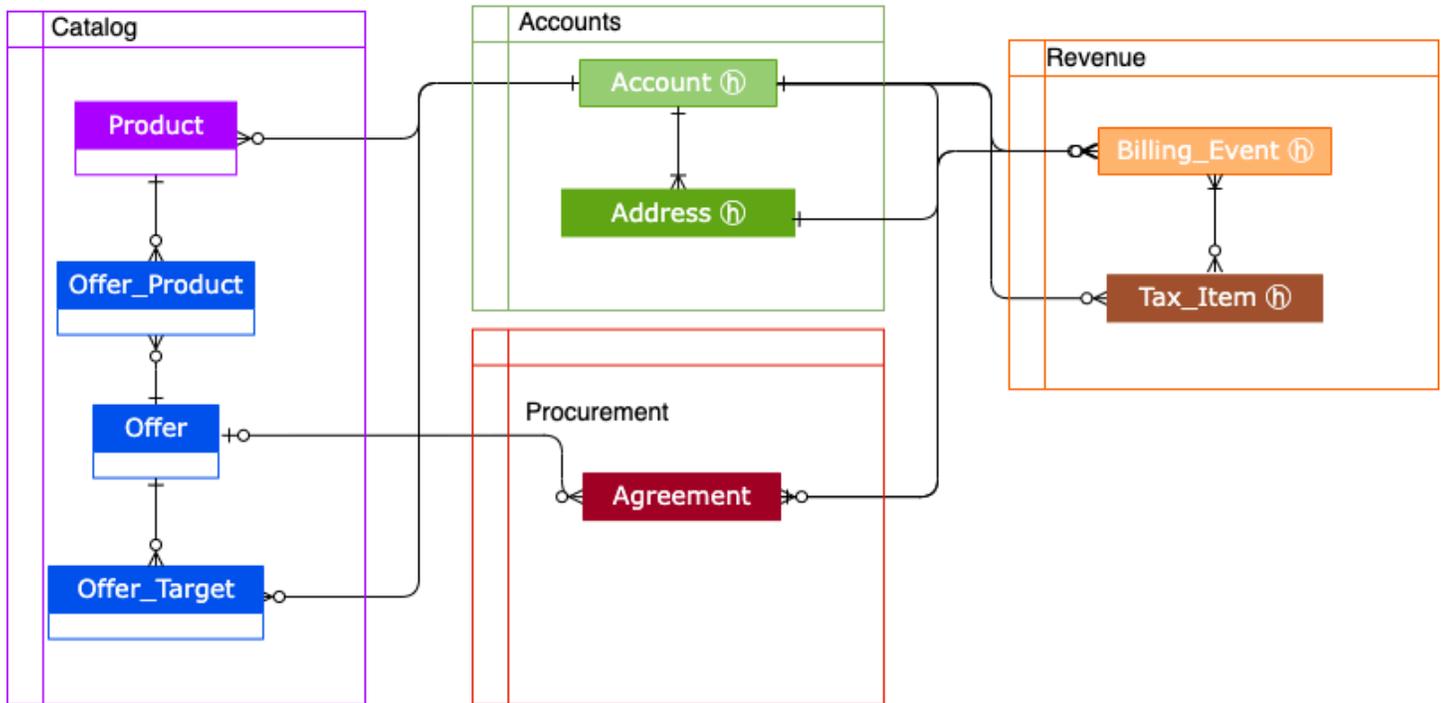
Ikhtisar tabel umpan data

AWS Marketplace menyediakan umpan data sebagai kumpulan tabel yang dapat Anda gabungkan bersama untuk memberikan lebih banyak konteks untuk kueri Anda.

AWS Marketplace menyediakan domain umum berikut, atau kategori yang menarik, dalam umpan data Anda:

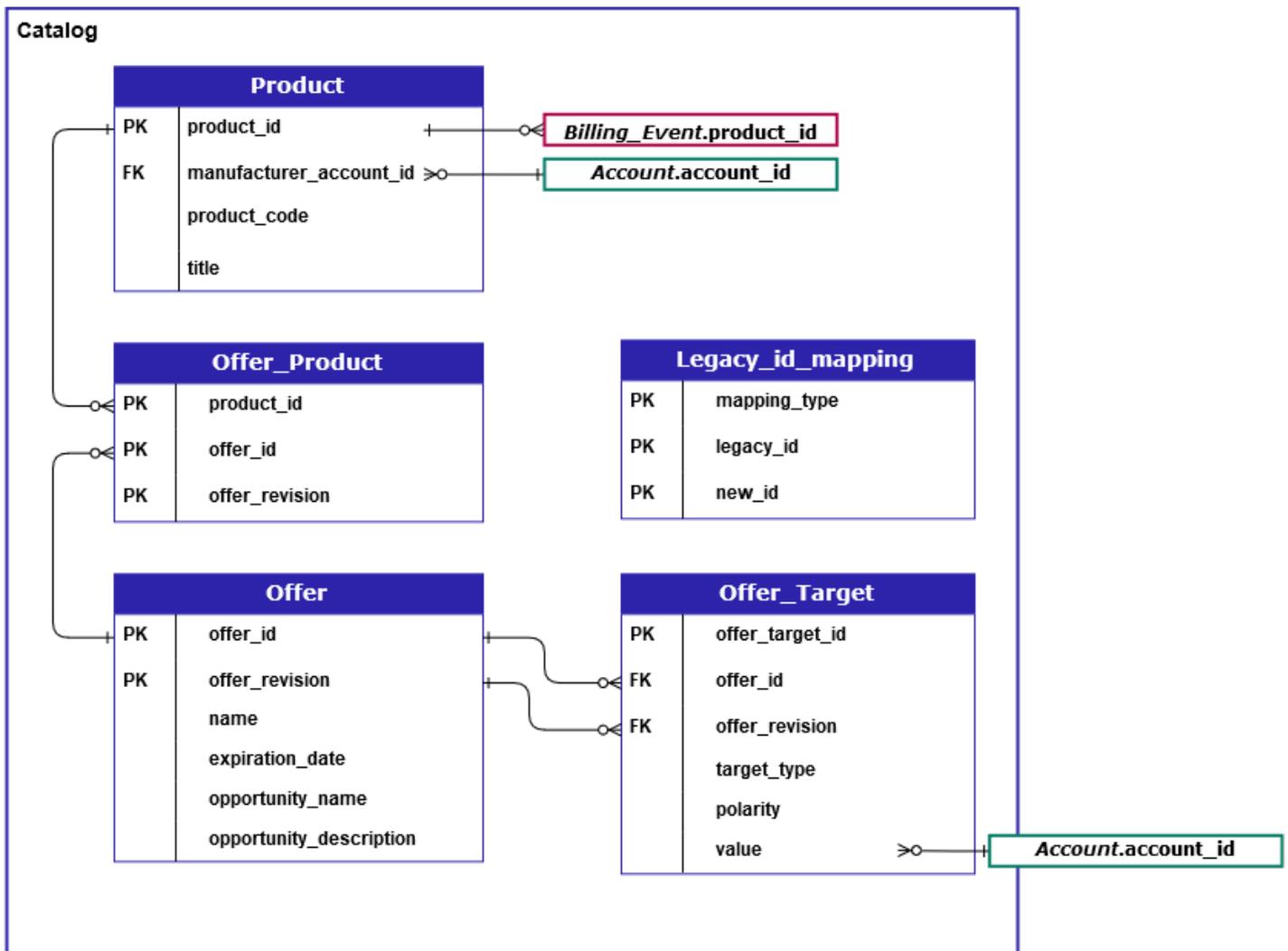
- Katalog - Termasuk informasi tentang produk dan penawaran di akun Anda.
- Akun — Termasuk informasi tentang akun yang menyediakan atau membeli produk di AWS Marketplace (akun Anda sendiri atau akun pihak yang bekerja dengan Anda seperti mitra saluran atau pembeli).
- Pendapatan - Termasuk informasi tentang penagihan, pencairan, dan pajak.
- Pengadaan — mencakup informasi tentang perjanjian untuk penawaran produk yang telah Anda buat sebagai penjual catatan.

Diagram ini menunjukkan tabel di domain Katalog, Akun, dan Pendapatan.



Tabel terkait katalog

Diagram berikut menunjukkan hubungan antara tabel dalam domain Katalog, serta bidang dalam tabel.



Legacy_id_mapping Tabel Product Offer_Product, Offer, Offer_Target, dan Account berada di domain Katalog.

Offer_Target Tabel mencakup bidang nilai untuk target, tetapi hanya jika target_type nilainya account_id.

Legacy_id_mapping Tabel tidak digunakan untuk data saat ini.

Note

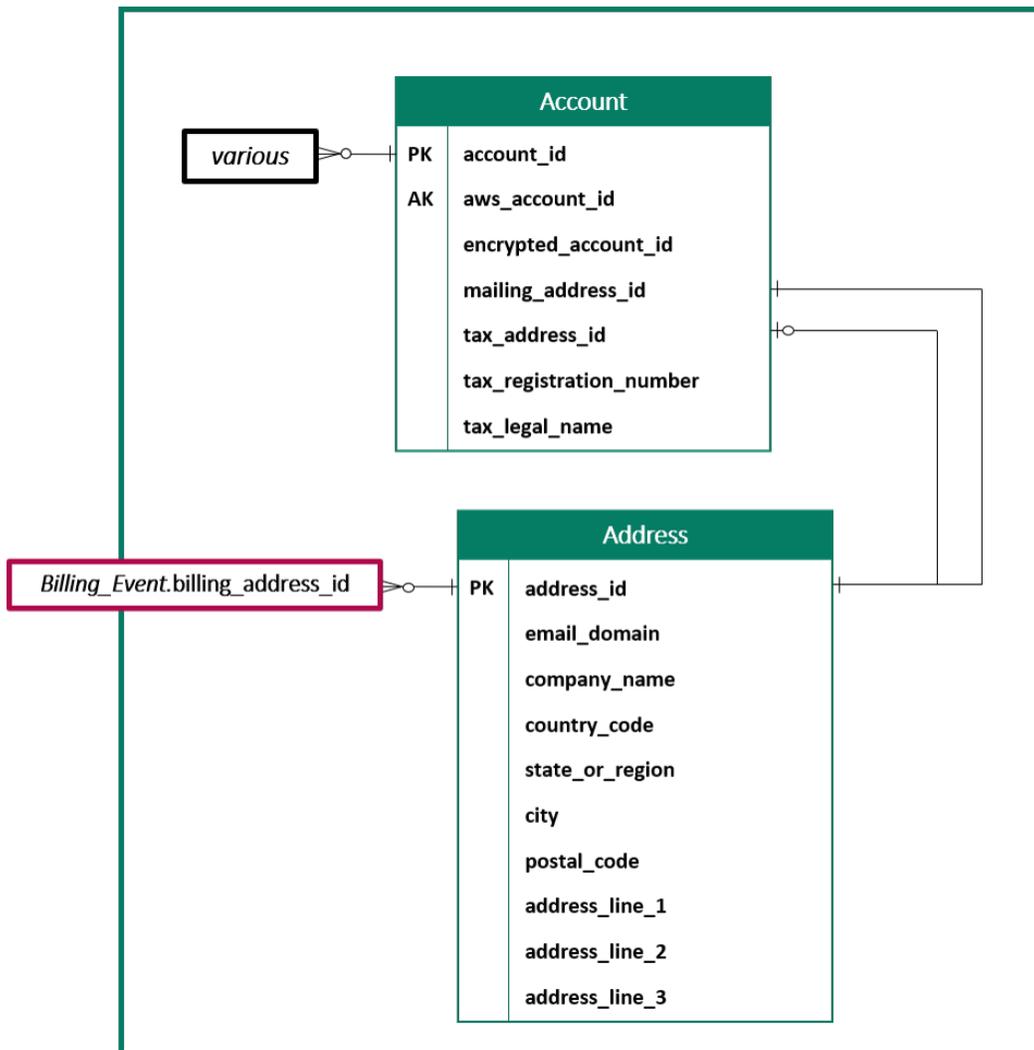
Untuk informasi selengkapnya tentang tabel ini, termasuk deskripsi setiap bidang dalam tabel dan gabungan yang dapat dibuat, lihat topik berikut:

- [Umpan data produk](#)
- [Tawarkan umpan data produk](#)

- [Tawarkan umpan data](#)
- [Tawarkan umpan data target](#)
- [Umpan data pemetaan lama](#)

Tabel terkait akun

Diagram berikut menunjukkan hubungan antara Account dan Address tabel dalam domain Accounts, serta bidang dalam tabel.

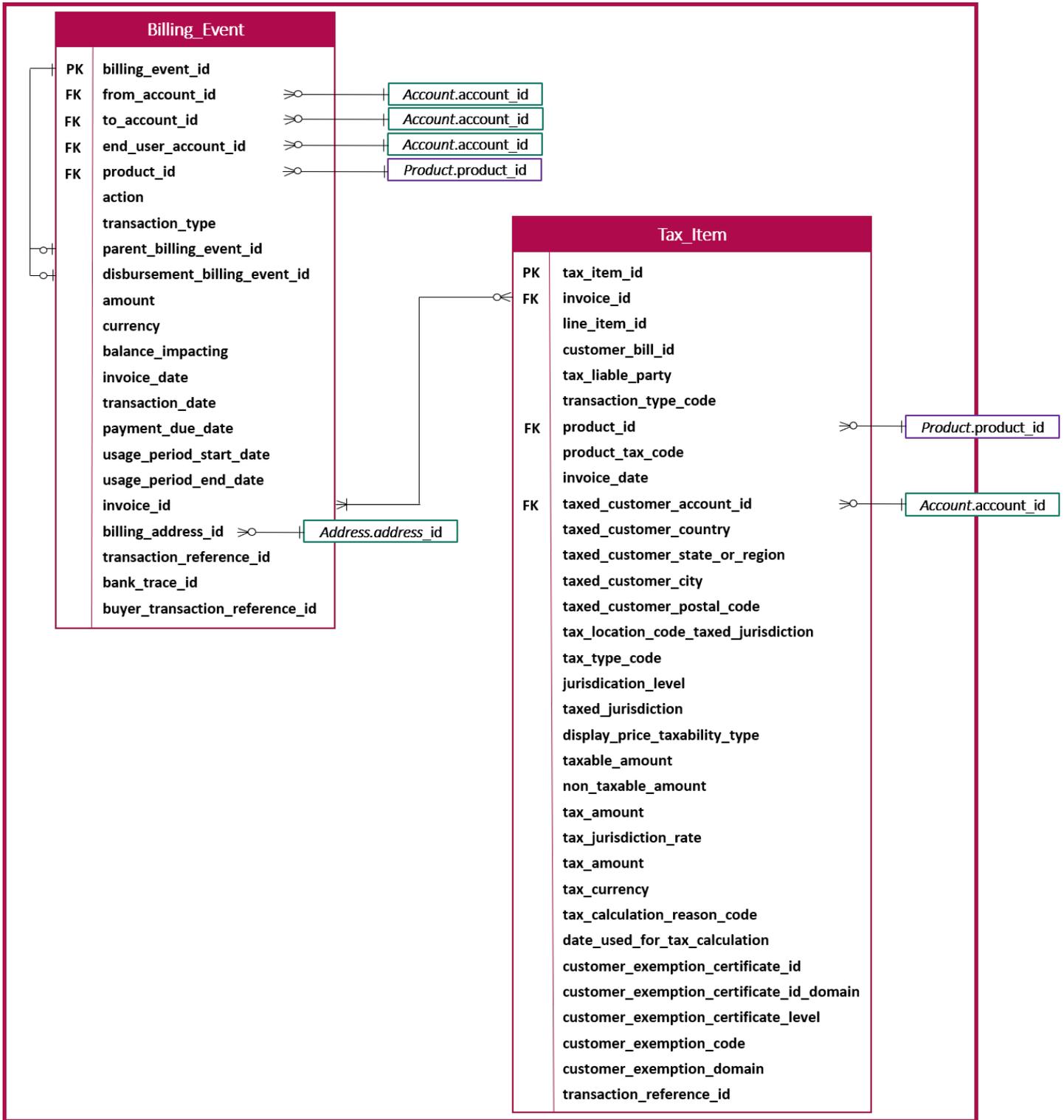


Note
 Untuk informasi selengkapnya tentang tabel ini, termasuk deskripsi setiap bidang dalam tabel dan gabungan yang dapat Anda buat, lihat topik berikut:

- [Umpan data akun](#)
- [Alamat umpan data](#)

Tabel terkait pendapatan

Diagram berikut menunjukkan hubungan antara `Billing_Event` dan `Tax_Item` tabel dalam domain Pendapatan, serta bidang dalam tabel. `Billing_Event` tabel ini mencakup informasi tentang pencairan, serta acara penagihan.



Note

Untuk informasi selengkapnya tentang tabel ini, termasuk deskripsi setiap bidang dalam tabel dan gabungan yang dapat Anda buat, lihat topik berikut:

- [Umpan data acara penagihan](#)
- [Umpan data item pajak](#)

Tabel terkait pengadaan

Diagram berikut menunjukkan bidang dalam tabel Perjanjian di domain Pengadaan.

Note

Untuk informasi selengkapnya tentang tabel ini, termasuk deskripsi setiap bidang dalam tabel dan gabungan yang dapat dibuat, lihat [Umpan data perjanjian](#), dalam panduan ini.

Bagian berikut menyediakan diagram relasi entitas (ER) untuk setiap domain. Setiap diagram ER menunjukkan tabel dan bidang dalam setiap tabel, serta bidang yang dapat Anda gunakan untuk bergabung dengan tabel.

Note

Diagram ER di bagian ini tidak menyertakan bidang umum untuk semua umpan data. Untuk informasi lebih lanjut tentang bidang umum, lihat [Penyimpanan dan struktur umpan AWS Marketplace data](#).

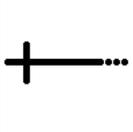
Tabel berikut menjelaskan simbol yang digunakan dalam diagram ER.

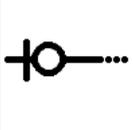
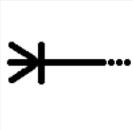
Simbol	Deskripsi
PK	Primary key — Kunci utama untuk tabel. Ketika digunakan dengan <code>valid_from</code> dan <code>update_date</code> bidang, itu unik. Untuk detail selengkapnya tentang menggunakan bidang ini bersama-sama, lihat Historisasi data . Jika lebih

Simbol	Deskripsi
	dari satu bidang ditandai sebagai kunci utama, maka bidang bersama-sama membentuk kunci utama.
	<p>Foreign key — Bidang yang mewakili kunci utama dalam tabel yang berbeda. Belum tentu unik di tabel.</p> <div style="border: 1px solid #add8e6; border-radius: 10px; padding: 10px; margin-top: 10px;"> <p>Note</p> <p>Dalam beberapa kasus, kunci asing dapat kosong jika catatan dalam tabel saat ini tidak memiliki catatan yang sesuai di tabel asing.</p> </div>
	<p>Tombol alternatif — Kunci yang dapat digunakan sebagai kunci dalam tabel. Mengikuti aturan keunikan yang sama dengan kunci utama.</p>
	<p>Konektor — Garis antar bidang mewakili koneksi, yang merupakan dua bidang yang dapat digunakan untuk menggabungkan tabel. Ujung garis mewakili jenis koneksi. Contoh ini mewakili one-to-many koneksi.</p>

Jenis konektor

Tabel berikut menunjukkan jenis ujung yang dapat dimiliki setiap konektor.

Jenis konektor	Deskripsi
	<p>Satu ke n — Konektor dengan ujung ini mewakili gabungan yang memiliki tepat satu nilai di sisi gabungan ini.</p>

Jenis konektor	Deskripsi
	Nol atau satu ke n — Konektor dengan ujung ini mewakili gabungan yang memiliki nol atau satu nilai di sisi gabungan ini.
	Nol atau lebih ke n — Konektor dengan ujung ini mewakili gabungan yang memiliki nol, satu, atau banyak nilai di sisi gabungan ini.
	Satu atau lebih ke n - Konektor dengan ujung ini mewakili gabungan yang memiliki satu atau banyak nilai di sisi gabungan ini.

Contoh kueri umpan data

Bagian ini memberikan contoh kueri kompleks menggunakan umpan data yang disediakan oleh AWS Marketplace. Contoh-contoh ini mirip dengan [Laporan penjual di AWS Marketplace](#) yang Anda dapatkan dari Portal Manajemen AWS Marketplace. Anda dapat menyesuaikan kueri ini untuk membuat laporan lain yang Anda butuhkan.

Kueri contoh

- [Perjanjian dan pembaruan](#)
- [Pendapatan yang ditagih](#)
- [Faktur yang tidak ditagih atau dicairkan](#)
- [Faktur pajak](#)
- [Pencairan berdasarkan produk](#)
- [Laporan kompensasi penjualan](#)

Perjanjian dan pembaruan

Untuk menemukan data perjanjian dan perpanjangan, Anda dapat menjalankan serangkaian kueri seperti contoh berikut. Kueri dibangun di atas satu sama lain untuk membuat dasbor Perjanjian dan pembaruan, bagian data terperinci. Anda dapat menggunakan contoh seperti yang ditunjukkan, atau menyesuaikannya untuk data dan kasus penggunaan Anda.

Komentar dalam kueri menjelaskan apa yang dilakukan kueri, dan bagaimana memodifikasinya.

```
Query currently under development.
```

Pendapatan yang ditagih

Untuk menemukan data faktur, Anda dapat menjalankan serangkaian kueri seperti contoh berikut. Kueri dibangun di atas satu sama lain untuk membuat laporan pendapatan yang ditagih. Anda dapat menggunakan contoh seperti yang ditunjukkan, atau menyesuaikannya untuk data dan kasus penggunaan Anda.

Komentar dalam kueri menjelaskan apa yang dilakukan kueri, dan bagaimana memodifikasinya.

```
-- Billed revenue report

-- General note: When executing this query we are assuming that the data ingested in
  the database uses
-- two time axes (the valid_from column and the update_date column).
-- See documentation for more details: https://docs.aws.amazon.com/marketplace/latest/userguide/data-feed.html#data-feed-details

-- An account_id has several valid_from dates (each representing a separate revision of
  the data)
-- but because of bi-temporality, an account_id + valid_from tuple can appear multiple
  times with a different update_date.
-- We are only interested in the most recent tuple (ie, uni-temporal model)
with accounts_with_uni_temporal_data as (
  select
    account_id,
    aws_account_id,
    encrypted_account_id,
    mailing_address_id,
    tax_address_id,
    tax_legal_name,
    from_iso8601_timestamp(valid_from) as valid_from,
    tax_registration_number
  from
    (
      select
        account_id,
        aws_account_id,
```

```

        encrypted_account_id,
        mailing_address_id,
        tax_address_id,
        tax_legal_name,
        valid_from,
        delete_date,
        tax_registration_number,
        row_number() over (partition by account_id, valid_from order by
from_iso8601_timestamp(update_date) desc) as row_num
    from
        accountfeed_v1
)
where
    -- keep latest ...
    row_num = 1
    -- ... and remove the soft-deleted one.
    and (delete_date is null or delete_date = '')
),

accounts_with_history as (
    with accounts_with_history_with_extended_valid_from as (
        select
            account_id,
            -- sometimes, this columns gets imported as a "bigint" and loses heading 0s ->
            casting to a char and re-adding heading 0s (if need be)
            substring('000000000000' || cast(aws_account_id as varchar), -12) as aws_account_id,
            encrypted_account_id,
            mailing_address_id,
            tax_address_id,
            tax_legal_name,
            -- The start time of account valid_from is extended to '1970-01-01 00:00:00',
because:
            -- ... in tax report transformations, some tax line items with invoice_date
cannot
            -- ... fall into the default valid time range of the associated account
            case
                when lag(valid_from) over (partition by account_id order by valid_from asc) is
null
                    then cast('1970-01-01 00:00:00' as timestamp)
                    else valid_from
            end as valid_from
        from accounts_with_uni_temporal_data
    )
select

```

```
account_id,  
aws_account_id,  
encrypted_account_id,  
mailing_address_id,  
tax_address_id,  
tax_legal_name,  
valid_from,  
coalesce(  
    lead(valid_from) over (partition by account_id order by valid_from asc),  
    cast('2999-01-01 00:00:00' as timestamp)  
) as valid_to  
from  
    accounts_with_history_with_extended_valid_from  
)  
  
-- An address_id has several valid_from dates (each representing a separate revision of  
the data)  
-- but because of bi-temporality, an account_id + valid_from tuple can appear multiple  
times with a different update_date.  
-- We are only interested in the most recent tuple (ie, uni-temporal model)  
address_with_uni_temporal_data as (  
    select  
        from_iso8601_timestamp(valid_from) as valid_from,  
        address_id,  
        company_name,  
        email_domain,  
        country_code,  
        state_or_region,  
        city,  
        postal_code,  
        row_num  
    from  
    (  
        select  
            valid_from,  
            update_date,  
            delete_date,  
            address_id,  
            company_name,  
            email_domain,  
            country_code,  
            state_or_region,  
            city,  
            postal_code,
```

```
    row_number() over (partition by address_id, valid_from order by
from_iso8601_timestamp(update_date) desc) as row_num
    from
        addressfeed_v1
    )
where
    -- keep latest ...
    row_num = 1
    -- ... and remove the soft-deleted one.
    and (delete_date is null or delete_date = '')
),

-- We are only interested in the most recent tuple (BTW: a given address is not
supposed to change over time but when bugs ;-) so this query mainly does nothing)
address_with_latest_revision as (
    select
        valid_from,
        address_id,
        company_name,
        email_domain,
        country_code,
        state_or_region,
        city,
        postal_code,
        row_num_latest_revision
    from
    (
        select
            valid_from,
            address_id,
            company_name,
            email_domain,
            country_code,
            state_or_region,
            city,
            postal_code,
            row_number() over (partition by address_id order by valid_from desc) as
row_num_latest_revision
        from
            address_with_uni_temporal_data
        )
    where
        row_num_latest_revision = 1
),
```

```

accounts_with_history_with_company_name as (
  select
    awh.account_id,
    awh.aws_account_id,
    awh.encrypted_account_id,
    awh.mailing_address_id,
    awh.tax_address_id,
    coalesce(
      --empty value in Athena shows as '', change all '' value to null
      case when address.company_name = '' then null else address.company_name end,
      awh.tax_legal_name) as mailing_company_name,
    address.email_domain,
    awh.valid_from,
    -- For BYOL, the agreement might be accepted (using some external non-AWS system or
    manual process) days before
    -- that BYOL agreement is entered into AWS Marketplace by the buyer. Therefore, the
    buyer is permitted to manually
    -- enter a backdated acceptance date, which might predate the point in time when
    the account was created.
    -- To work around this, we need to adjust the valid_from of the account to be
    -- earlier than the earliest possible backdated BYOL agreement acceptance date.
    case
      when lag(awh.valid_from) over (partition by aws_account_id order by
awh.valid_from asc) is null
      then date_add('Day', -212, awh.valid_from)
      -- 212 is the longest delay between acceptance_date of the agreement and the
account start_Date
      else awh.valid_from
    end as valid_from_adjusted,
    awh.valid_to
  from accounts_with_history as awh
  left join address_with_latest_revision as address on
    awh.mailing_address_id = address.address_id and awh.mailing_address_id is not null
),

-- An agreement_id has several valid_from dates (each representing an agreement
revision)
-- but because of bi-temporality, an agreement_id + valid_from tuple can appear
multiple times with a different update_date.
-- We are only interested in the most recent tuple (ie, uni-temporal model)
agreements_with_uni_temporal_data as (
  select
    agreement_id,

```

```

    origin_offer_id,
    proposer_account_id,
    acceptor_account_id,
    agreement_revision,
    from_iso8601_timestamp(valid_from) as valid_from,
    from_iso8601_timestamp(start_date) as start_date,
    from_iso8601_timestamp(end_date) as end_date,
    from_iso8601_timestamp(acceptance_date) as acceptance_date,
    agreement_type,
    previous_agreement_id,
    agreement_intent
from
(
  select
    --empty value in Athena shows as '', change all '' value to null
    case when agreement_id = '' then null else agreement_id end as agreement_id,
    origin_offer_id,
    proposer_account_id,
    acceptor_account_id,
    agreement_revision,
    valid_from,
    delete_date,
    start_date,
    end_date,
    acceptance_date,
    agreement_type,
    previous_agreement_id,
    agreement_intent,
    row_number() over (partition by agreement_id, valid_from order by
from_iso8601_timestamp(update_date) desc) as row_num
  from
    -- TODO change to agreementfeed_v1 when Agreement Feed is GA'ed
    agreementfeed
)
where
  -- keep latest ...
  row_num = 1
  -- ... and remove the soft-deleted one.
  and (delete_date is null or delete_date = '')
),

agreements_with_history as (
  with agreements_with_window_functions as (
    select

```

```

    agreement_id,
    origin_offer_id as offer_id,
    proposer_account_id,
    acceptor_account_id,
    agreement_revision,
    start_date,
    end_date,
    acceptance_date,
    -- The start time of agreement valid_from is extended to '1970-01-01 00:00:00',
because:
    -- ... in usage report transformations, some usage line items with usage_date
cannot
    -- ... fall into the default valid time range of the associated agreement
case
    when lag(valid_from) over (PARTITION BY agreement_id order by valid_from asc)
is null
    then timestamp '1970-01-01 00:00:00'
    else valid_from
end as valid_from,
coalesce(
    lead(valid_from) over (partition by agreement_id order by valid_from asc),
    timestamp '2999-01-01 00:00:00'
) as valid_to,
rank() over (partition by agreement_id order by valid_from asc) version,
agreement_type,
previous_agreement_id,
agreement_intent
from
    agreements_with_uni_temporal_data
)
select
    agreement_id,
    offer_id,
    proposer_account_id,
    acceptor_account_id,
    agreement_revision,
    start_date,
    end_date,
    acceptance_date,
    valid_from,
    case
        when version=1 and valid_from<timestamp '2023-03-03 06:16:08.743' then
timestamp '1970-01-01'

```

```
-- The following 60 minute adjustment is to handle special case where When
Renewal happens for a contract
    when version=1 then date_add('minute',-60,valid_from)
    else valid_from
end as valid_from_adjusted,
valid_to,
agreement_type,
previous_agreement_id,
agreement_intent
from
    agreements_with_window_functions
),

-- An offer_id has several valid_from dates (each representing an offer revision)
-- but because of bi-temporality, an offer_id + valid_from tuple can appear multiple
times with a different update_date.
-- We are only interested in the most recent tuple (ie, uni-temporal model)
offers_with_uni_temporal_data as (
    select
        from_iso8601_timestamp(valid_from) as valid_from,
        from_iso8601_timestamp(update_date) as update_date,
        from_iso8601_timestamp(delete_date) as delete_date,
        offer_id,
        offer_revision,
        name,
        expiration_date,
        opportunity_id,
        opportunity_name,
        opportunity_description,
        seller_account_id
    from
        (
            select
                valid_from,
                update_date,
                delete_date,
                offer_id,
                offer_revision,
                name,
                expiration_date,
                opportunity_id,
                opportunity_name,
                opportunity_description,
                seller_account_id,
```

```

        row_number() over (partition by offer_id, valid_from order by
from_iso8601_timestamp(update_date) desc) as row_num
    from
        offerfeed_v1
    )
where
    -- keep latest ...
    row_num = 1
    -- ... and remove the soft-deleted one.
    and (delete_date is null or delete_date = '')
),

-- Here, we build the validity time range (adding valid_to on top of valid_from) of
each offer revision.
-- We will use it to get Offer name at invoice time.
-- NB: If you'd rather get "current" offer name, un-comment
"offers_with_latest_revision"
offers_with_history as (
    select
        offer_id,
        offer_revision,
        name,
        opportunity_id,
        opportunity_name,
        opportunity_description,
        valid_from,
        -- When we try to look up an offer revision as at the acceptance date of a BYOL
agreement, we run into a problem.
        -- For BYOL, the agreement might be accepted (using some external non-AWS system or
manual process) days before
        -- that BYOL agreement is entered into AWS Marketplace by the buyer. Therefore, the
buyer is permitted to manually
        -- enter a backdated acceptance date, which might predate the point in time when
the first revision of the offer
        -- was created. To work around this, we need to adjust the valid_from on the first
revision of the offer to be
        -- earlier than the earliest possible backdated BYOL agreement acceptance date.
        case
            when lag(valid_from) over (partition by offer_id order by valid_from asc) is null
and valid_from < cast('2021-04-01' as timestamp)
            then date_add('Day', -3857, valid_from)
            -- 3857 is the longest delay between acceptance_date of an agreement and the
first revision of the offer

```

```
    when lag(valid_from) over (partition by offer_id order by valid_from asc) is null
and valid_from >= cast('2021-04-01' as timestamp)
    then date_add('Day', -1460, valid_from)
    --after 2021 for the two offers we need to adjust for 2 more years
    else valid_from
end as valid_from_adjusted,
coalesce(
    lead(valid_from) over (partition by offer_id order by valid_from asc),
    cast('2999-01-01 00:00:00' as timestamp))
as valid_to
from offers_with_uni_temporal_data
),
-- provided for reference only if you are interested into get "current" offer name
-- (ie. not used afterwards)
offers_with_latest_revision as (
    select
        offer_id,
        offer_revision,
        name,
        opportunity_name,
        opportunity_description,
        valid_from,
        null valid_to
    from
    (
        select
            offer_id,
            offer_revision,
            name,
            opportunity_name,
            opportunity_description,
            valid_from,
            null valid_to,
            row_number() over (partition by offer_id order by valid_from desc) as
row_num_latest_revision
        from
            offers_with_uni_temporal_data
    )
    where
        row_num_latest_revision = 1
),

-- An offer_target_id has several valid_from dates (each representing an offer
revision)
```

```
-- but because of bi-temporality, an offer_target_id + valid_from tuple can appear
multiple times with a different update_date.
-- We are only interested in the most recent tuple (ie, uni-temporal model)
offer_targets_with_uni_temporal_data as (
  select
    from_iso8601_timestamp(valid_from) as valid_from,
    from_iso8601_timestamp(update_date) as update_date,
    from_iso8601_timestamp(delete_date) as delete_date,
    offer_target_id,
    offer_id,
    offer_revision,
    target_type,
    polarity,
    value
  from
    (
      select
        valid_from,
        update_date,
        delete_date,
        offer_target_id,
        offer_id,
        offer_revision,
        target_type,
        polarity,
        value,
        row_number() over (partition by offer_target_id, valid_from order by
from_iso8601_timestamp(update_date) desc) as row_num
      from
        offertargetfeed_v1
    )
  where
    -- keep latest ...
    row_num = 1
    -- ... and remove the soft-deleted one.
    and (delete_date is null or delete_date = '')
),

offer_target_type as (
  select
    offer_id,
    offer_revision,
    substring(
```

```

-- The first character indicates the priority (lower value means higher
precedence):
min(
  case
    when offer_target.target_type='BuyerAccounts' then '1Private'
    when offer_target.target_type='ParticipatingPrograms' then '2Program:'||
cast(offer_target.value as varchar)
    when offer_target.target_type='CountryCodes' then '3GeoTargeted'
    -- well, there is no other case today, but rather be safe...
    else '4Other Targeting'
  end
),
-- Remove the first character that was only used for th priority in the "min"
aggregate function:
2
) as offer_target
from
  offer_targets_with_uni_temporal_data as offer_target
group by
  offer_id,
  offer_revision
),
offers_with_history_with_target_type as (
  select
    offer.offer_id,
    offer.offer_revision,
    -- even though today it is not possible to combine several types of targeting in a
single offer, let's ensure the query is still predictable if this gets possible in the
future
    max(
      case
        when off_tgt.offer_target is null then 'Public'
        else off_tgt.offer_target
      end
    ) as offer_target,
    min(offer.name) as name,
    min(offer.opportunity_name) as opportunity_name,
    min(offer.opportunity_description) as opportunity_description,
    offer.valid_from,
    offer.valid_from_adjusted,
    offer.valid_to,
    offer.opportunity_id
  from

```

```

    offers_with_history as offer
left join offer_target_type as off_tgt on
    offer.offer_id = off_tgt.offer_id
    and offer.offer_revision = off_tgt.offer_revision
group by
    offer.offer_id,
    offer.offer_revision,
    offer.valid_from,
    offer.valid_from_adjusted,
    offer.valid_to,
    offer.opportunity_id
),

-- provided for reference only if you are interested into get "current" offer targets
-- (ie. not used afterwards)
offers_with_latest_revision_with_target_type as (
    select
        offer.offer_id,
        offer.offer_revision,
        -- even though today it is not possible to combine several types of targeting in a
        single offer, let's ensure the query is still predictable if this gets possible in the
        future
        max(
            distinct
            case
                when off_tgt.target_type is null then 'Public'
                when off_tgt.target_type='BuyerAccounts' then 'Private'
                when off_tgt.target_type='ParticipatingPrograms' then 'Program:'||
cast(off_tgt.value as varchar)
                when off_tgt.target_type='CountryCodes' then 'GeoTargeted'
                -- well, there is no other case today, but rather be safe...
                else 'Other Targeting'
            end
        ) as offer_target,
        min(offer.name) as name,
        min(offer.opportunity_name) as opportunity_name,
        min(offer.opportunity_description) as opportunity_description,
        offer.valid_from,
        offer.valid_to
    from
        offers_with_latest_revision offer
        -- left joining because public offers don't have targets
    left join offer_targets_with_uni_temporal_data off_tgt on
        offer.offer_id=off_tgt.offer_id and offer.offer_revision=off_tgt.offer_revision

```

```
group by
    offer.offer_id,
    offer.offer_revision,
    -- redundant with offer_revision, as each revision has a dedicated valid_from (but
cleaner in the group by)
    offer.valid_from,
    offer.valid_to
),

-- A product_id has several valid_from dates (each representing a product revision),
-- but because of bi-temporality, each product_id + valid_from tuple can appear
multiple times with a different update_date.
-- We are only interested in the most recent tuple (ie, uni-temporal model)
products_with_uni_temporal_data as (
    select
        from_iso8601_timestamp(valid_from) as valid_from,
        from_iso8601_timestamp(update_date) as update_date,
        from_iso8601_timestamp(delete_date) as delete_date,
        product_id,
        manufacturer_account_id,
        product_code,
        title
    from
    (
        select
            valid_from,
            update_date,
            delete_date,
            product_id,
            manufacturer_account_id,
            product_code,
            title,
            row_number() over (partition by product_id, valid_from order by
from_iso8601_timestamp(update_date) desc) as row_num
        from
            productfeed_v1
    )
    where
        -- keep latest ...
        row_num = 1
        -- ... and remove the soft-deleted one.
        and (delete_date is null or delete_date = '')
),
```

```
products_with_history as (  
  select  
    product_id,  
    title,  
    valid_from,  
    -- Offererv2 can have upto 50 years and Offererv3 is upto 5 years of past date  
    case  
      when lag(valid_from) over (partition by product_id order by valid_from asc) is  
      null and valid_from < cast('2021-04-01' as timestamp)  
        then date_add('Day', -3857, valid_from)  
        -- 3827 is the longest delay between acceptance_date of an agreement and the  
product  
        -- we are keeping 3857 as a consistency between the offers and products  
      when lag(valid_from) over (partition by product_id order by valid_from asc) is  
      null and valid_from >= cast('2021-04-01' as timestamp)  
        then date_add('Day', -2190, valid_from)  
        --after 2021 for the two offers we need to adjust for 2 more years  
      else valid_from  
    end as valid_from_adjusted,  
    coalesce(  
      lead(valid_from) over (partition by product_id order by valid_from asc),  
      cast('2999-01-01 00:00:00' as timestamp)  
    ) as valid_to,  
    product_code,  
    manufacturer_account_id  
  from  
    products_with_uni_temporal_data  
)  
  
legacy_products as (  
  select  
    legacy_id,  
    new_id  
  from  
    legacyidmappingfeed_v1  
  where  
    mapping_type='PRODUCT'  
  group by  
    legacy_id,  
    new_id  
)  
  
-- A given billing_event_id represents an accounting event and thus has only one  
valid_from date,
```

```
-- but because of bi-temporality, a billing_event_id (+ its valid_from) can appear
multiple times with a different update_date.
-- We are only interested in the most recent tuple (ie, uni-temporal model)
billing_events_with_uni_temporal_data as (
  select
    billing_event_id,
    valid_from,
    update_date,
    delete_date,
    invoice_date,
    transaction_type,
    transaction_reference_id,
    parent_billing_event_id,
    bank_trace_id,
    broker_id,
    product_id,
    disbursement_billing_event_id,
    action,
    from_account_id,
    to_account_id,
    end_user_account_id,
    billing_address_id,
    amount,
    currency,
    balance_impacting,
    --empty value in Athena shows as '', change all '' value to null
    case when agreement_id = '' then null else agreement_id end as agreement_id,
    invoice_id,
    payment_due_date,
    usage_period_start_date,
    usage_period_end_date,
    buyer_transaction_reference_id,
    row_num
  from
    (
      select
        billing_event_id,
        from_iso8601_timestamp(valid_from) as valid_from,
        from_iso8601_timestamp(update_date) as update_date,
        delete_date,
        from_iso8601_timestamp(invoice_date) as invoice_date,
        transaction_type,
        transaction_reference_id,
        parent_billing_event_id,
```

```

-- casting in case data was imported as number
cast(bank_trace_id as varchar) as bank_trace_id,
broker_id,
product_id,
disbursement_billing_event_id,
action,
from_account_id,
to_account_id,
end_user_account_id,
billing_address_id,
-- casting in case data was imported as varchar
cast(amount as decimal(38,6)) as amount,
currency,
balance_impacting,
agreement_id,
invoice_id,
case when payment_due_date is null or payment_due_date = '' then null else
from_iso8601_timestamp(payment_due_date) end as payment_due_date,
from_iso8601_timestamp(usage_period_start_date) as usage_period_start_date,
from_iso8601_timestamp(usage_period_end_date) as usage_period_end_date,
buyer_transaction_reference_id,
row_number() over (partition by billing_event_id, valid_from order by
from_iso8601_timestamp(update_date) desc) as row_num
from
    billingeventfeed_v1
)
where
    -- keep latest ...
    row_num = 1
    -- ... and remove the soft-deleted one.
    and (delete_date is null or delete_date = '')
),

-- Here we select the account_id of the current seller (We identify this by looking for
the to_account_id related to revenue transactions).
-- We will use it later to distinguish own agreements from agreements generated by
channel partners.
seller_account as (
    select
        from_account_id as seller_account_id
    from
        billing_events_with_uni_temporal_data bill
    where

```

```

-- Assumption here is only seller will pay listing fee. As of 12/21/2021, there are
cases that Channel partner have 0 listing fee for CPP0, so the amount could be 0.
bill.transaction_type like 'AWS_REV_SHARE' and amount <= 0 and action = 'INVOICED'
group by
-- from_account_id is always the same for all those "listing fee" transactions ==
the seller of record himself.
-- If this view returns more than 1 record, the overall query will fail (on
purpose). Please contact AWS Marketplace if this happens.
from_account_id
),

```

```

billing_event_with_business_flags as (
select
  bl.billing_event_id,
  bl.end_user_account_id,
  bl.agreement_id,
  agreement.proposer_account_id,
  agreement.offer_id,
  agreement.acceptor_account_id,
  case
    -- For AWS and BALANCE_ADJUSTMENT, the billing event feed will show the "AWS
Marketplace" account as the
    -- receiver of the funds and the seller as the payer. We are not interested in
this information here.
    -- Null values will be ignored by the `max` aggregation function.
    when bl.transaction_type like 'AWS%' then null
    -- For BALANCE_ADJUSTMENT, payer is seller themselves
    when bl.invoice_id is null then bl.to_account_id
    -- We get the payer of the invoice from *any* transaction type that is not AWS
and not BALANCE_ADJUSTMENT (because they are the same for a given end user + agreement
+ product).
    else bl.from_account_id
  end as payer_account_id,
  bl.product_id,
  bl.action,
  bl.transaction_type,
  bl.parent_billing_event_id,
  bl.disbursement_billing_event_id,
  bl.amount,
  bl.currency,
  bl.balance_impacting,
  bl.invoice_date,
  bl.payment_due_date,
  bl.usage_period_start_date,

```

```

bl.usage_period_end_date,
bl.invoice_id,
bl.billing_address_id,
bl.transaction_reference_id,
bl.buyer_transaction_reference_id,
case when disbursement.bank_trace_id = 'EMEA_MP_TEST_TRACE_ID' then null else
disbursement.bank_trace_id end as bank_trace_id,
case when disbursement.bank_trace_id = 'EMEA_MP_TEST_TRACE_ID' then null else
disbursement.invoice_date end as disbursement_date,
disbursement.billing_event_id as disbursement_id,
-- We will use disbursement_id_or_invoiced as part of the PK, so it cannot be null:
coalesce(
--empty value in Athena shows as '', change all '' value to null
case when disbursement.billing_event_id = '' then null else
disbursement.billing_event_id end,
'<invoiced>') as disbursement_id_or_invoiced,
bl.broker_id,
case
when bl.invoice_id is null /* transaction_type = 'BALANCE_ADJUSTMENT' */
then (select seller_account_id from seller_account) ||':'||
cast(bl.invoice_date as varchar)
else bl.buyer_transaction_reference_id
||'-'|| case when bl.agreement_id is null or bl.agreement_id = '' then ' ' else
bl.agreement_id end
||'-'|| case when bl.end_user_account_id is null or bl.end_user_account_id = ''
then ' ' else bl.end_user_account_id end
||'-'|| coalesce(cast(bl.usage_period_start_date as varchar),' ')
||'-'|| coalesce(cast(bl.usage_period_end_date as varchar),' ')
end as internal_buyer_line_item_id,
bl.buyer_transaction_reference_id <> bl.transaction_reference_id as
is_seller_invoice,
case when bl.transaction_type = 'SELLER_REV_SHARE' and (select seller_account_id
from seller_account) <> bl.to_account_id then true else false end as is_cog,
case when bl.transaction_type in('SELLER_REV_SHARE_CREDIT',
'SELLER_REV_SHARE_REFUND') and (select seller_account_id from seller_account) <>
bl.to_account_id then true else false end as is_cog_refund,
--TODO: replace below logic once we can create a logic the identify reseller/
manufacturer without depending on agreement feed
case when agreement.proposer_account_id <> (select seller_account_id from
seller_account) then true else false end as is_manufacturer_view_of_reseller
from
billing_events_with_uni_temporal_data as bl
left join billing_events_with_uni_temporal_data as disbursement on
disbursement.transaction_type like 'DISBURSEMENT%'

```

```

        and disbursement.action = 'DISBURSED'
        and disbursement.transaction_type IN ('DISBURSEMENT', 'DISBURSEMENT_FAILURE')
        and bl.disbursement_billing_event_id = disbursement.billing_event_id
left join agreements_with_history as aggrement on
    bl.agreement_id = aggrement.agreement_id
    and bl.invoice_date >= aggrement.valid_from_adjusted
    and bl.invoice_date < aggrement.valid_to
left join accounts_with_history awh on
    bl.to_account_id = awh.account_id
    and bl.invoice_date >= awh.valid_from
    and bl.invoice_date < awh.valid_to
where
    bl.transaction_type not like 'DISBURSEMENT%' and
    (bl.agreement_id is null or bl.agreement_id = ''
    or aggrement.agreement_id is not null)
),

-- listagg function in athena does not support partitioning, grouping here and then
  joining to the main query
seller_invoice_list as (
  select
    internal_buyer_line_item_id,
    listagg(case when not is_seller_invoice then null else invoice_id end, ',') within
group (order by case when not is_seller_invoice then null else cast(invoice_date as
varchar) end) as seller_invoice_id_or_null,
    listagg(case when not is_seller_invoice then null else cast(invoice_date as
varchar) end, ',') within group (order by case when not is_seller_invoice then null
else cast(invoice_date as varchar) end) as seller_invoice_date_or_null
  from
    (
      -- listagg function in athena does not support ordering by another field when
distinct is used,
      -- here we first select distinct invoices and then do the listagg order by
invoice_date
      select distinct internal_buyer_line_item_id, is_seller_invoice, invoice_id,
invoice_date
      from billing_event_with_business_flags) distinct_invoices
  group by internal_buyer_line_item_id
  order by internal_buyer_line_item_id
),

billing_event_with_categorized_transaction as (
-- Use the flags that were created in the previous transformation in more calculated
columns:

```

```
-- NOTE: This transformation has no joins and no window functions
select
  billing_event_id,
  end_user_account_id,
  agreement_id,
  proposer_account_id,
  offer_id,
  acceptor_account_id,
  case when is_cog or is_cog_refund then null else payer_account_id end as
payer_account_id,
  product_id,
  action,
  transaction_type,
  parent_billing_event_id,
  disbursement_billing_event_id,
  amount,
  currency,
  balance_impacting,
  invoice_date,
  payment_due_date,
  usage_period_start_date,
  usage_period_end_date,
  invoice_id,
  billing_address_id,
  transaction_reference_id,
  buyer_transaction_reference_id,
  bank_trace_id,
  disbursement_date,
  disbursement_id,
  disbursement_id_or_invoiced,
  broker_id,
  bl.internal_buyer_line_item_id,
  is_seller_invoice,
  is_cog,
  is_cog_refund,
  is_manufacturer_view_of_reseller,

  -- Buyer/seller columns:
  case when is_seller_invoice then null else invoice_id end as
buyer_invoice_id_or_null,
  seller_invoices.seller_invoice_id_or_null,
  case when is_seller_invoice then null else invoice_date end as
buyer_invoice_date_or_null,
  seller_invoices.seller_invoice_date_or_null,
```

```

-- Categorized amounts by transaction type:
case when transaction_type = 'SELLER_REV_SHARE' and not is_cog then amount else 0
end as gross_revenue,
case when transaction_type in ('SELLER_REV_SHARE_REFUND','SELLER_REV_SHARE_CREDIT')
and not is_cog_refund then amount else 0 end as gross_refund,
case when transaction_type = 'SELLER_REV_SHARE' and is_cog then amount else 0
end as cogs,
case when transaction_type in ('SELLER_REV_SHARE_REFUND','SELLER_REV_SHARE_CREDIT')
and is_cog_refund then amount else 0 end as cogs_refund,
case when transaction_type = 'AWS_REV_SHARE' then amount else 0 end as
aws_rev_share,
case when transaction_type in ('AWS_REV_SHARE_REFUND','AWS_REV_SHARE_CREDIT') then
amount else 0 end as aws_refund_share,
case when transaction_type = 'AWS_TAX_SHARE' and not is_seller_invoice then
amount else 0 end as aws_tax_share, -- AWS tax share from_buyer_ invoice
case when transaction_type = 'AWS_TAX_SHARE' and is_seller_invoice then
amount else 0 end as aws_tax_share_listing_fee, -- AWS tax share from_seller_ invoice
case when transaction_type = 'AWS_TAX_SHARE_REFUND' and not is_seller_invoice
then amount else 0 end as aws_tax_share_refund,
case when transaction_type = 'AWS_TAX_SHARE_REFUND' and is_seller_invoice
then amount else 0 end as aws_tax_share_refund_listing_fee,
case when transaction_type = 'SELLER_TAX_SHARE' then amount else 0 end as
seller_tax_share,
case when transaction_type = 'SELLER_TAX_SHARE_REFUND' then amount else 0 end as
seller_tax_share_refund,
case when transaction_type = 'BALANCE_ADJUSTMENT' then amount else 0 end as
balance_adjustment,
case when transaction_type = 'SELLER_REV_SHARE_CREDIT' then amount else 0 end as
seller_rev_credit,
case when transaction_type = 'AWS_REV_SHARE_CREDIT' then amount else 0 end as
aws_ref_fee_credit
from
  billing_event_with_business_flags as bl
  left join seller_invoice_list as seller_invoices
    on bl.internal_buyer_line_item_id = seller_invoices.internal_buyer_line_item_id
),

line_items_aggregated as (
-- This transformation has the only "group by" in all of these transformations.
-- NOTE: This transformation has no joins and no window functions
select
  internal_buyer_line_item_id,
  disbursement_id,

```

```

disbursement_id_or_invoiced,
product_id,
broker_id,
currency,
agreement_id,
proposer_account_id,
acceptor_account_id,
max(payer_account_id) as payer_account_id,
offer_id,
end_user_account_id,
usage_period_start_date,
usage_period_end_date,
max(payment_due_date) payment_due_date,
buyer_transaction_reference_id,
bank_trace_id,
disbursement_date,
max(billing_address_id) as billing_address_id,

-- Buyer/seller columns:
max(buyer_invoice_id_or_null) as buyer_invoice_id,
max(seller_invoice_id_or_null) as seller_invoice_id,
max(buyer_invoice_date_or_null) as buyer_invoice_date,
max(seller_invoice_date_or_null) as seller_invoice_date,

-- Categorized amounts by transaction type:
-- When disbursement_id_or_invoiced = '<invoiced>', these are invoiced amounts
-- When disbursement_id_or_invoiced <> '<invoiced>' these are disbursed amounts for
_this_ specific disbursement_id
sum(gross_revenue) as gross_revenue_this_disbursement_id_or_invoiced,
sum(gross_refund) as gross_refund_this_disbursement_id_or_invoiced,
sum(cogs) as cogs_this_disbursement_id_or_invoiced,
sum(cogs_refund) as cogs_refund_this_disbursement_id_or_invoiced,
sum(aws_rev_share) as aws_rev_share_this_disbursement_id_or_invoiced,
sum(aws_refund_share) as aws_refund_share_this_disbursement_id_or_invoiced,
sum(aws_tax_share) as aws_tax_share_this_disbursement_id_or_invoiced,
sum(aws_tax_share_listing_fee) as
aws_tax_share_listing_fee_this_disbursement_id_or_invoiced,
sum(aws_tax_share_refund) as aws_tax_share_refund_this_disbursement_id_or_invoiced,
sum(aws_tax_share_refund_listing_fee) as
aws_tax_share_refund_listing_fee_this_disbursement_id_or_invoiced,
sum(seller_tax_share) as seller_tax_share_this_disbursement_id_or_invoiced,
sum(seller_tax_share_refund) as
seller_tax_share_refund_this_disbursement_id_or_invoiced,
sum(balance_adjustment) as balance_adjustment_this_disbursement_id_or_invoiced,

```

```

    sum(seller_rev_credit) as seller_rev_credit_this_disbursement_id_or_invoiced,
    sum(aws_ref_fee_credit) as aws_ref_fee_credit_this_disbursement_id_or_invoiced
from
    billing_event_with_categorized_transaction as billing_categorized
group by
    internal_buyer_line_item_id,
    disbursement_id,
    disbursement_id_or_invoiced,
    broker_id,
    -- The following columns are included the in group by but they are intentionally
omitted from the PK.
    -- These columns should have the _same_ values for each record in the PK.
    product_id,
    currency,
    agreement_id,
    proposer_account_id,
    acceptor_account_id,
    offer_id,
    end_user_account_id,
    usage_period_start_date,
    usage_period_end_date,
    buyer_transaction_reference_id,
    bank_trace_id,
    disbursement_date
),
-- listagg function in athena does not support partitioning, grouping here and then
joining to the main query
disbursement_list as (
    select
        internal_buyer_line_item_id,
        listagg(cast(disbursement_date as varchar),',') within group (order by
cast(disbursement_date as varchar)) as disbursement_date_list,
        listagg(bank_trace_id,',') within group (order by cast(disbursement_date as
varchar)) as disburse_bank_trace_id_list
    from (
        -- listagg function in athena does not support ordering by another field when
distinct is used,
        -- here we first select distinct bank_trace_ids and then do the listagg order by
disbursement_date
        select distinct internal_buyer_line_item_id, disbursement_date, bank_trace_id
from billing_event_with_business_flags) distinct_disbursements
    group by internal_buyer_line_item_id
    order by internal_buyer_line_item_id

```

```
),  
  
line_items_with_window_functions as (  
--add flag next step compare gross_revenue and gross_revenue_disbursed or gross_refund  
and gross_refund_disbursed  
select  
    line_item.internal_buyer_line_item_id,  
    disbursement_id,  
    disbursement_id_or_invoiced,  
    product_id,  
    broker_id,  
    currency,  
    agreement_id,  
    proposer_account_id,  
    acceptor_account_id,  
    -- when there's aws_rev_Share adjustment/refund to a seller_rev_share invoice, it  
    can happen that for the same aws_rev_share invoice_id, there are multiple disbursement  
    events,  
    -- using windows function to map payer_account_id of seller_rev_share to all  
    corresponding aws_rev_Share  
    max(payer_account_id) over (partition by line_item.internal_buyer_line_item_id) as  
    payer_account_id,  
    offer_id,  
    end_user_account_id,  
    usage_period_start_date,  
    usage_period_end_date,  
    payment_due_date,  
    bank_trace_id,  
    disbursement_date,  
    billing_address_id,  
  
    -- Buyer/seller columns:  
    max(buyer_invoice_id) over (partition by line_item.internal_buyer_line_item_id) as  
    buyer_invoice_id,  
    seller_invoice_id,  
    max(buyer_invoice_date) over (partition by line_item.internal_buyer_line_item_id)  
    as buyer_invoice_date,  
    seller_invoice_date,  
  
    -- When disbursement_id_or_invoiced = '<invoiced>', these are actually invoiced  
    amounts  
    -- When disbursement_id_or_invoiced <> '<invoiced>' these are disbursed amounts for  
    _this_ specific disbursement_id  
    gross_revenue_this_disbursement_id_or_invoiced,
```

```

gross_refund_this_disbursement_id_or_invoiced,
cogs_this_disbursement_id_or_invoiced,
cogs_refund_this_disbursement_id_or_invoiced,
aws_rev_share_this_disbursement_id_or_invoiced,
aws_refund_share_this_disbursement_id_or_invoiced,
aws_tax_share_this_disbursement_id_or_invoiced,
aws_tax_share_listing_fee_this_disbursement_id_or_invoiced,
aws_tax_share_refund_this_disbursement_id_or_invoiced,
aws_tax_share_refund_listing_fee_this_disbursement_id_or_invoiced,
seller_tax_share_this_disbursement_id_or_invoiced,
seller_tax_share_refund_this_disbursement_id_or_invoiced,
balance_adjustment_this_disbursement_id_or_invoiced,
seller_rev_credit_this_disbursement_id_or_invoiced,
aws_ref_fee_credit_this_disbursement_id_or_invoiced,
-- IMPORTANT: All window functions partitioned by internal_buyer_line_item_id:

-- Invoiced amounts, categorized by transaction type:
sum(case when disbursement_id_or_invoiced = '<invoiced>' then
gross_revenue_this_disbursement_id_or_invoiced else cast(0 as decimal(38,6)) end)over
(partition by line_item.internal_buyer_line_item_id) as gross_revenue_invoiced,
sum(case when disbursement_id_or_invoiced = '<invoiced>' then
gross_refund_this_disbursement_id_or_invoiced else cast(0 as decimal(38,6)) end) over
(partition by line_item.internal_buyer_line_item_id) as gross_refund_invoiced,
sum(case when disbursement_id_or_invoiced = '<invoiced>' then
cogs_this_disbursement_id_or_invoiced else cast(0 as decimal(38,6)) end) over
(partition by line_item.internal_buyer_line_item_id) as cogs_invoiced,
sum(case when disbursement_id_or_invoiced = '<invoiced>' then
cogs_refund_this_disbursement_id_or_invoiced else cast(0 as decimal(38,6)) end) over
(partition by line_item.internal_buyer_line_item_id) as cogs_refund_invoiced,
sum(case when disbursement_id_or_invoiced = '<invoiced>' then
aws_rev_share_this_disbursement_id_or_invoiced else cast(0 as decimal(38,6)) end) over
(partition by line_item.internal_buyer_line_item_id) as aws_rev_share_invoiced,
sum(case when disbursement_id_or_invoiced = '<invoiced>' then
aws_refund_share_this_disbursement_id_or_invoiced else cast(0 as decimal(38,6))
end) over (partition by line_item.internal_buyer_line_item_id) as
aws_refund_share_invoiced,
sum(case when disbursement_id_or_invoiced = '<invoiced>' then
aws_tax_share_this_disbursement_id_or_invoiced else cast(0 as decimal(38,6)) end) over
(partition by line_item.internal_buyer_line_item_id) as aws_tax_share_invoiced,
sum(case when disbursement_id_or_invoiced = '<invoiced>' then
aws_tax_share_listing_fee_this_disbursement_id_or_invoiced else cast(0 as
decimal(38,6)) end) over (partition by line_item.internal_buyer_line_item_id) as
aws_tax_share_listing_fee_invoiced,

```

```

sum(case when disbursement_id_or_invoiced = '<invoiced>' then
aws_tax_share_refund_this_disbursement_id_or_invoiced else cast(0 as
decimal(38,6)) end) over (partition by line_item.internal_buyer_line_item_id) as
aws_tax_share_refund_invoiced,
sum(case when disbursement_id_or_invoiced = '<invoiced>' then
aws_tax_share_refund_listing_fee_this_disbursement_id_or_invoiced else cast(0 as
decimal(38,6)) end) over (partition by line_item.internal_buyer_line_item_id) as
aws_tax_share_refund_listing_fee_invoiced,
sum(case when disbursement_id_or_invoiced = '<invoiced>' then
seller_tax_share_this_disbursement_id_or_invoiced else cast(0 as decimal(38,6))
end) over (partition by line_item.internal_buyer_line_item_id) as
seller_tax_share_invoiced,
sum(case when disbursement_id_or_invoiced = '<invoiced>' then
seller_tax_share_refund_this_disbursement_id_or_invoiced else cast(0 as
decimal(38,6)) end) over (partition by line_item.internal_buyer_line_item_id) as
seller_tax_share_refund_invoiced,
sum(case when disbursement_id_or_invoiced = '<invoiced>' then
balance_adjustment_this_disbursement_id_or_invoiced else cast(0 as
decimal(38,6)) end) over (partition by line_item.internal_buyer_line_item_id) as
balance_adjustment_invoiced,
sum(case when disbursement_id_or_invoiced = '<invoiced>' then
seller_rev_credit_this_disbursement_id_or_invoiced else cast(0 as
decimal(38,6)) end) over (partition by line_item.internal_buyer_line_item_id) as
seller_rev_credit_invoiced,
sum(case when disbursement_id_or_invoiced = '<invoiced>' then
aws_ref_fee_credit_this_disbursement_id_or_invoiced else cast(0 as
decimal(38,6)) end) over (partition by line_item.internal_buyer_line_item_id) as
aws_ref_fee_credit_invoiced,

-- Total disbursed amounts (for all disbursement_id values), categorized by
transaction type:
sum(case when disbursement_id_or_invoiced <> '<invoiced>' then
gross_revenue_this_disbursement_id_or_invoiced else cast(0 as decimal(38,6)) end) over
(partition by line_item.internal_buyer_line_item_id) as gross_revenue_disbursed,
sum(case when disbursement_id_or_invoiced <> '<invoiced>' then
gross_refund_this_disbursement_id_or_invoiced else cast(0 as decimal(38,6)) end) over
(partition by line_item.internal_buyer_line_item_id) as gross_refund_disbursed,
sum(case when disbursement_id_or_invoiced <> '<invoiced>' then
cogs_this_disbursement_id_or_invoiced else cast(0 as decimal(38,6)) end) over
(partition by line_item.internal_buyer_line_item_id) as cogs_disbursed,
sum(case when disbursement_id_or_invoiced <> '<invoiced>' then
cogs_refund_this_disbursement_id_or_invoiced else cast(0 as decimal(38,6)) end) over
(partition by line_item.internal_buyer_line_item_id) as cogs_refund_disbursed,

```

```
sum(case when disbursement_id_or_invoiced <> '<invoiced>' then
aws_rev_share_this_disbursement_id_or_invoiced else cast(0 as decimal(38,6)) end) over
(partition by line_item.internal_buyer_line_item_id) as aws_rev_share_disbursed,
sum(case when disbursement_id_or_invoiced <> '<invoiced>' then
aws_refund_share_this_disbursement_id_or_invoiced else cast(0 as decimal(38,6))
end) over (partition by line_item.internal_buyer_line_item_id) as
aws_refund_share_disbursed,
sum(case when disbursement_id_or_invoiced <> '<invoiced>' then
aws_tax_share_this_disbursement_id_or_invoiced else cast(0 as decimal(38,6)) end) over
(partition by line_item.internal_buyer_line_item_id) as aws_tax_share_disbursed,
sum(case when disbursement_id_or_invoiced <> '<invoiced>' then
aws_tax_share_listing_fee_this_disbursement_id_or_invoiced else cast(0 as
decimal(38,6)) end) over (partition by line_item.internal_buyer_line_item_id) as
aws_tax_share_listing_fee_disbursed,
sum(case when disbursement_id_or_invoiced <> '<invoiced>' then
aws_tax_share_refund_this_disbursement_id_or_invoiced else cast(0 as
decimal(38,6)) end) over (partition by line_item.internal_buyer_line_item_id) as
aws_tax_share_refund_disbursed,
sum(case when disbursement_id_or_invoiced <> '<invoiced>' then
aws_tax_share_refund_listing_fee_this_disbursement_id_or_invoiced else cast(0 as
decimal(38,6)) end) over (partition by line_item.internal_buyer_line_item_id) as
aws_tax_share_refund_listing_fee_disbursed,
sum(case when disbursement_id_or_invoiced <> '<invoiced>' then
seller_tax_share_this_disbursement_id_or_invoiced else cast(0 as decimal(38,6))
end) over (partition by line_item.internal_buyer_line_item_id) as
seller_tax_share_disbursed,
sum(case when disbursement_id_or_invoiced <> '<invoiced>' then
seller_tax_share_refund_this_disbursement_id_or_invoiced else cast(0 as
decimal(38,6)) end) over (partition by line_item.internal_buyer_line_item_id) as
seller_tax_share_refund_disbursed,
sum(case when disbursement_id_or_invoiced <> '<invoiced>' then
balance_adjustment_this_disbursement_id_or_invoiced else cast(0 as
decimal(38,6)) end) over (partition by line_item.internal_buyer_line_item_id) as
balance_adjustment_disbursed,
sum(case when disbursement_id_or_invoiced <> '<invoiced>' then
seller_rev_credit_this_disbursement_id_or_invoiced else cast(0 as
decimal(38,6)) end) over (partition by line_item.internal_buyer_line_item_id) as
seller_rev_credit_disbursed,
sum(case when disbursement_id_or_invoiced <> '<invoiced>' then
aws_ref_fee_credit_this_disbursement_id_or_invoiced else cast(0 as
decimal(38,6)) end) over (partition by line_item.internal_buyer_line_item_id) as
aws_ref_fee_credit_disbursed,

-- aggregate multiple disbursement
```

```

    max(disbursement_date) over (partition by line_item.internal_buyer_line_item_id) as
last_disbursement_date,
    first_value(case when disbursement_id_or_invoiced =
'<invoiced>' then null else disbursement_id_or_invoiced end)
over(partition by line_item.internal_buyer_line_item_id order by
coalesce(disbursement_date,cast('1900-01-01' as timestamp)) desc rows between
unbounded preceding and unbounded following) as last_disbursement_id,
    first_value(bank_trace_id) over (partition by line_item.internal_buyer_line_item_id
order by coalesce(disbursement_date,cast('1900-01-01' as timestamp)) desc rows between
unbounded preceding and unbounded following) as last_disburse_bank_trace_id,
    disb_list.disbursement_date_list,
    disb_list.disburse_bank_trace_id_list
from
    line_items_aggregated as line_item
    left join disbursement_list disb_list
        on line_item.internal_buyer_line_item_id = disb_list.internal_buyer_line_item_id
),

cpo_offer_id as (
    select
        -- Channel partner offers do not exist in offertargetfeed_v1 table (as per legal
requirement), causing cppo offer be defined as 'Public' in previous step, we will
convert them back to 'Private' in next step
        offer_id
    from
        offers_with_uni_temporal_data
    where
        -- seller_account_id is null means the ISV owns the offer
        seller_account_id is not null
        and seller_account_id <> (select seller_account_id from seller_account)
    group by
        offer_id
),

line_items_with_window_functions_enrich_offer_product_address as (
    select
        internal_buyer_line_item_id,
        disbursement_id,
        disbursement_id_or_invoiced,
        line.product_id,
        legacy_product.legacy_id as legacy_product_id,
        products.title as product_title,
        line.broker_id,
        line.currency,

```

```
line.end_user_account_id,  
acc_enduser.encrypted_account_id as end_user_encrypted_account_id,  
acc_enduser.aws_account_id as end_user_aws_account_id,  
acc_payer.aws_account_id as payer_aws_account_id,  
acc_payer.encrypted_account_id payer_encrypted_account_id,  
line.agreement_id,  
agreement.agreement_revision,  
line.proposer_account_id,  
case when offer.offer_id like 'aiqoffer-%' then null else agreement.start_date end  
as Agreement_Start_Date,  
case when offer.offer_id like 'aiqoffer-%' then null else agreement.end_date end as  
Agreement_End_Date,  
case when offer.offer_id like 'aiqoffer-%' then null else agreement.acceptance_date  
end as Agreement_Acceptance_Date,  
case when offer.offer_id like 'aiqoffer-%' then null else agreement.valid_from end  
as agreement_updated_date,  
case when offer.offer_id like 'aiqoffer-%' then null else  
line.usage_period_start_date end as Usage_Period_Start_Date,  
case when offer.offer_id like 'aiqoffer-%' then null else  
line.usage_period_end_date end as Usage_Period_End_Date,  
  
line.acceptor_account_id,  
acc_subscriber.aws_account_id as subscriber_aws_account_id,  
acc_subscriber.encrypted_account_id as subscriber_encrypted_account_id,  
offer.offer_id,  
case  
  when offer.offer_id in (  
    select distinct offer_id  
    from cppo_offer_id)  
  then 'Private'  
  else offer.offer_target  
end as offer_target,  
offer.name offer_name,  
offer.opportunity_name offer_opportunity_name,  
offer.opportunity_description offer_opportunity_description,  
offer.opportunity_id,  
payment_due_date,  
line.bank_trace_id,  
disbursement_date,  
billing_address_id,  
buyer_invoice_id,  
seller_invoice_id,  
buyer_invoice_date,  
seller_invoice_date,
```

```
gross_revenue_this_disbursement_id_or_invoiced,  
gross_refund_this_disbursement_id_or_invoiced,  
cogs_this_disbursement_id_or_invoiced,  
cogs_refund_this_disbursement_id_or_invoiced,  
aws_rev_share_this_disbursement_id_or_invoiced,  
aws_refund_share_this_disbursement_id_or_invoiced,  
aws_tax_share_this_disbursement_id_or_invoiced,  
aws_tax_share_listing_fee_this_disbursement_id_or_invoiced,  
aws_tax_share_refund_this_disbursement_id_or_invoiced,  
aws_tax_share_refund_listing_fee_this_disbursement_id_or_invoiced,  
seller_tax_share_this_disbursement_id_or_invoiced,  
seller_tax_share_refund_this_disbursement_id_or_invoiced,  
balance_adjustment_this_disbursement_id_or_invoiced,  
seller_rev_credit_this_disbursement_id_or_invoiced,  
aws_ref_fee_credit_this_disbursement_id_or_invoiced,  
gross_revenue_invoiced,  
gross_refund_invoiced,  
cogs_invoiced,  
cogs_refund_invoiced,  
aws_rev_share_invoiced,  
aws_refund_share_invoiced,  
aws_tax_share_invoiced,  
aws_tax_share_listing_fee_invoiced,  
aws_tax_share_refund_invoiced,  
aws_tax_share_refund_listing_fee_invoiced,  
seller_tax_share_invoiced,  
seller_tax_share_refund_invoiced,  
balance_adjustment_invoiced,  
seller_rev_credit_invoiced,  
aws_ref_fee_credit_invoiced,  
gross_revenue_disbursed,  
gross_refund_disbursed,  
cogs_disbursed,  
cogs_refund_disbursed,  
aws_rev_share_disbursed,  
aws_refund_share_disbursed,  
aws_tax_share_disbursed,  
aws_tax_share_listing_fee_disbursed,  
aws_tax_share_refund_disbursed,  
aws_tax_share_refund_listing_fee_disbursed,  
seller_tax_share_disbursed,  
seller_tax_share_refund_disbursed,  
balance_adjustment_disbursed,  
seller_rev_credit_disbursed,
```

```

aws_ref_fee_credit_disbursed,
last_disbursement_date,
last_disbursement_id,
last_disburse_bank_trace_id,
disbursement_date_list,
disburse_bank_trace_id_list,
products.product_code,
acc_products.aws_account_id as manufacturer_aws_account_id,
products.manufacturer_account_id,
--add subscriber and payer addressID, payer address preference order: tax address >
billing address > mailing address, subscriber address preference order: tax address >
mailing address
coalesce (
  --empty value in Athena shows as '', change all '' value to null in order to
follow the preference order logic above
  case when acc_subscriber.tax_address_id = '' then null else
acc_subscriber.tax_address_id end,
  case when acc_subscriber.mailing_address_id = '' then null else
acc_subscriber.mailing_address_id end) as subscriber_address_id,
coalesce (
  case when acc_payer.tax_address_id = '' then null else acc_payer.tax_address_id
end,
  case when line.billing_address_id = '' then null else line.billing_address_id
end,
  case when acc_payer.mailing_address_id = '' then null else
acc_payer.mailing_address_id end) as payer_address_id,
coalesce (
  case when acc_enduser.tax_address_id = '' then null else
acc_enduser.tax_address_id end,
  case when line.billing_address_id = '' then null else line.billing_address_id
end,
  case when acc_enduser.mailing_address_id = '' then null else
acc_enduser.mailing_address_id end) as end_user_address_id
from
  line_items_with_window_functions as line
left join agreements_with_history as agreement on
  line.agreement_id = agreement.agreement_id and line.buyer_invoice_date >=
agreement.valid_from_adjusted and line.buyer_invoice_date < agreement.valid_to
left join offers_with_history_with_target_type as offer on
  line.offer_id = offer.offer_id and line.buyer_invoice_date >= offer.valid_from
and line.buyer_invoice_date < offer.valid_to
left join products_with_history as products on
  line.product_id = products.product_id and line.buyer_invoice_date >=
products.valid_from_adjusted and line.buyer_invoice_date < products.valid_to

```

```
left join legacy_products as legacy_product on
    line.product_id = legacy_product.new_id
left join accounts_with_history_with_company_name as acc_payer on
    line.payer_account_id = acc_payer.account_id and line.buyer_invoice_date >=
acc_payer.valid_from and line.buyer_invoice_date<acc_payer.valid_to
left join accounts_with_history_with_company_name as acc_enduser on
    line.end_user_account_id = acc_enduser.account_id and line.buyer_invoice_date
>= acc_enduser.valid_from and line.buyer_invoice_date<acc_enduser.valid_to
left join accounts_with_history_with_company_name as acc_subscriber on
    line.acceptor_account_id = acc_subscriber.account_id
and line.buyer_invoice_date >= acc_subscriber.valid_from and
line.buyer_invoice_date<acc_subscriber.valid_to
left join accounts_with_history_with_company_name as acc_products on
    products.manufacturer_account_id = acc_products.account_id
and line.buyer_invoice_date >= acc_products.valid_from and
line.buyer_invoice_date<acc_products.valid_to

),

line_items_with_window_functions_enrich_offer_product_address_name as (
select
    line.internal_buyer_line_item_id,
    disbursement_id,
    disbursement_id_or_invoiced,
    product_id,
    legacy_product_id,
    product_title,
    broker_id,
    currency,
    end_user_address_id,
    end_user_account_id,
    end_user_encrypted_account_id,
    end_user_aws_account_id,
    add_enduser.company_name end_user_company_name,
    add_enduser.email_domain end_user_email_domain,
    add_enduser.city end_user_city,
    add_enduser.state_or_region end_user_state,
    add_enduser.country_code end_user_country,
    add_enduser.postal_code end_user_postal_code,
    payer_aws_account_id,
    payer_encrypted_account_id,
    payer_address_id,
    add_payer.company_name payer_company_name,
    add_payer.email_domain payer_email_domain,
```

```
add_payer.city payer_city,
add_payer.state_or_region payer_state,
add_payer.country_code payer_country,
add_payer.postal_code payer_postal_code,
agreement_id,
agreement_revision,
agreement_start_date,
agreement_end_date,
agreement_acceptance_date,
agreement_updated_date,
case when proposer_account_id = (select seller_account_id from seller_account) then
null else acc_proposer.aws_account_id end as reseller_aws_account_id,
case when proposer_account_id = (select seller_account_id from seller_account) then
null else acc_proposer.mailing_company_name end as reseller_company_name,
usage_period_start_date,
usage_period_end_date,
proposer_account_id,
acc_proposer.aws_account_id as proposer_aws_account_id,
acceptor_account_id,
subscriber_aws_account_id,
subscriber_encrypted_account_id,
subscriber_address_id,
add_subscriber.company_name subscriber_company_name,
add_subscriber.email_domain subscriber_email_domain,
add_subscriber.city subscriber_city,
add_subscriber.state_or_region subscriber_state,
add_subscriber.country_code subscriber_country,
add_subscriber.postal_code subscriber_postal_code,
offer_id,
offer_target,
offer_name,
offer_opportunity_name,
offer_opportunity_description,
opportunity_id,
payment_due_date,
bank_trace_id,
disbursement_date,
billing_address_id,
max(buyer_invoice_id)as buyer_invoice_id,
max(seller_invoice_id)as seller_invoice_id,
max(buyer_invoice_date)as buyer_invoice_date,
max(seller_invoice_date)as seller_invoice_date,
gross_revenue_this_disbursement_id_or_invoiced,
gross_refund_this_disbursement_id_or_invoiced,
```

```
cogs_this_disbursement_id_or_invoiced,  
cogs_refund_this_disbursement_id_or_invoiced,  
aws_rev_share_this_disbursement_id_or_invoiced,  
aws_refund_share_this_disbursement_id_or_invoiced,  
aws_tax_share_this_disbursement_id_or_invoiced,  
aws_tax_share_listing_fee_this_disbursement_id_or_invoiced,  
aws_tax_share_refund_this_disbursement_id_or_invoiced,  
aws_tax_share_refund_listing_fee_this_disbursement_id_or_invoiced,  
seller_tax_share_this_disbursement_id_or_invoiced,  
seller_tax_share_refund_this_disbursement_id_or_invoiced,  
balance_adjustment_this_disbursement_id_or_invoiced,  
seller_rev_credit_this_disbursement_id_or_invoiced,  
aws_ref_fee_credit_this_disbursement_id_or_invoiced,  
(gross_revenue_this_disbursement_id_or_invoiced +  
gross_refund_this_disbursement_id_or_invoiced +  
aws_rev_share_this_disbursement_id_or_invoiced +  
aws_refund_share_this_disbursement_id_or_invoiced +  
seller_tax_share_this_disbursement_id_or_invoiced +  
seller_tax_share_refund_this_disbursement_id_or_invoiced  
+ cogs_this_disbursement_id_or_invoiced +  
cogs_refund_this_disbursement_id_or_invoiced +  
aws_tax_share_listing_fee_this_disbursement_id_or_invoiced +  
aws_tax_share_refund_listing_fee_this_disbursement_id_or_invoiced) as  
seller_net_revenue_this_disbursement_id_or_invoiced,  
gross_revenue_invoiced,  
gross_refund_invoiced,  
cogs_invoiced,  
cogs_refund_invoiced,  
aws_rev_share_invoiced,  
aws_refund_share_invoiced,  
aws_tax_share_invoiced,  
aws_tax_share_listing_fee_invoiced,  
aws_tax_share_refund_invoiced,  
aws_tax_share_refund_listing_fee_invoiced,  
seller_tax_share_invoiced,  
seller_tax_share_refund_invoiced,  
balance_adjustment_invoiced,  
seller_rev_credit_invoiced,  
aws_ref_fee_credit_invoiced,  
gross_revenue_disbursed,  
gross_refund_disbursed,  
cogs_disbursed,  
cogs_refund_disbursed,  
aws_rev_share_disbursed,
```

```

aws_refund_share_disbursed,
aws_tax_share_disbursed,
aws_tax_share_listing_fee_disbursed,
aws_tax_share_refund_disbursed,
aws_tax_share_refund_listing_fee_disbursed,
seller_tax_share_disbursed,
seller_tax_share_refund_disbursed,
balance_adjustment_disbursed,
seller_rev_credit_disbursed,
aws_ref_fee_credit_disbursed,
(gross_revenue_invoiced + gross_revenue_disbursed) as uncollected_gross_revenue,
-- net revenue = gross revenue - listing fee - tax - cogs
(gross_revenue_invoiced + gross_refund_invoiced + aws_rev_share_invoiced
+ aws_refund_share_invoiced + seller_tax_share_invoiced +
seller_tax_share_refund_invoiced + cogs_invoiced + cogs_refund_invoiced +
aws_tax_share_listing_fee_invoiced + aws_tax_share_refund_listing_fee_invoiced) as
seller_net_revenue,
(gross_revenue_invoiced + gross_refund_invoiced + aws_rev_share_invoiced
+ aws_refund_share_invoiced + seller_tax_share_invoiced +
seller_tax_share_refund_invoiced + cogs_invoiced + cogs_refund_invoiced +
aws_tax_share_listing_fee_invoiced + aws_tax_share_refund_listing_fee_invoiced
+ gross_revenue_disbursed + gross_refund_disbursed + aws_rev_share_disbursed
+ aws_refund_share_disbursed + seller_tax_share_disbursed +
seller_tax_share_refund_disbursed + cogs_disbursed + cogs_refund_disbursed +
aws_tax_share_listing_fee_disbursed + aws_tax_share_refund_listing_fee_disbursed) as
uncollected_seller_net_revenue,
last_disbursement_date,
last_disbursement_id,
last_disburse_bank_trace_id,
disbursement_date_list,
disburse_bank_trace_id_list,
product_code,
manufacturer_aws_account_id,
manufacturer_account_id,
acc_manu.mailing_company_name as manufacturer_company_name,
cast(null as varchar) as AR_Period,
case
  when (
    (gross_revenue_invoiced <> 0 and gross_revenue_invoiced = -1 *
gross_revenue_disbursed)
    or (gross_refund_invoiced <> 0 and gross_refund_invoiced = -1 *
gross_refund_disbursed)
    or (balance_adjustment_invoiced <> 0 and balance_adjustment_invoiced = -1 *
balance_adjustment_disbursed)

```

```

        or (seller_tax_share_refund_invoiced <> 0 and seller_tax_share_refund_invoiced
= -1 * seller_tax_share_refund_disbursed)
        or (gross_revenue_invoiced = 0 and gross_refund_invoiced = 0 and
balance_adjustment_invoiced = 0 and seller_tax_share_refund_invoiced = 0 and
last_disbursement_id is not null)) then 'Yes'
        when gross_revenue_disbursed = 0 and gross_refund_disbursed = 0 and
balance_adjustment_disbursed = 0 and seller_tax_share_disbursed = 0 and
seller_tax_share_refund_disbursed = 0 then 'No'
        else 'Partial'
    end as Disbursement_Flag
from line_items_with_window_functions_enrich_offer_product_address as line
left join accounts_with_history_with_company_name as acc_manu on
    line.manufacturer_account_id = acc_manu.account_id and line.buyer_invoice_date >=
acc_manu.valid_from_adjusted and line.buyer_invoice_date <= acc_manu.valid_to
left join accounts_with_history_with_company_name as acc_proposer on
    line.proposer_account_id = acc_proposer.account_id and line.buyer_invoice_date >=
acc_proposer.valid_from and line.buyer_invoice_date < acc_proposer.valid_to
left join address_with_latest_revision as add_payer on
    line.payer_address_id = add_payer.address_id
left join address_with_latest_revision as add_subscriber on
    line.subscriber_address_id = add_subscriber.address_id
left join address_with_latest_revision as add_enduser on
    line.end_user_address_id = add_enduser.address_id
group by
    line.internal_buyer_line_item_id,
    disbursement_id,
    disbursement_id_or_invoiced,
    product_id,
    legacy_product_id,
    product_title,
    broker_id,
    currency,
    end_user_address_id,
    end_user_account_id,
    end_user_encrypted_account_id,
    end_user_aws_account_id,
    add_enduser.company_name,
    add_enduser.email_domain,
    add_enduser.city,
    add_enduser.state_or_region,
    add_enduser.country_code,
    add_enduser.postal_code,
    payer_aws_account_id,
    payer_encrypted_account_id,

```

```
payer_address_id,  
add_payer.company_name,  
add_payer.email_domain,  
add_payer.city,  
add_payer.state_or_region,  
add_payer.country_code,  
add_payer.postal_code,  
agreement_id,  
agreement_revision,  
case when proposer_account_id = (select seller_account_id from seller_account) then  
null else acc_proposer.aws_account_id end,  
case when proposer_account_id = (select seller_account_id from seller_account) then  
null else acc_proposer.mailing_company_name end,  
agreement_start_date,  
agreement_end_date,  
agreement_acceptance_date,  
agreement_updated_date,  
usage_period_start_date,  
usage_period_end_date,  
acceptor_account_id,  
subscriber_aws_account_id,  
subscriber_encrypted_account_id,  
subscriber_address_id,  
add_subscriber.company_name,  
add_subscriber.email_domain,  
add_subscriber.city,  
add_subscriber.state_or_region,  
add_subscriber.country_code,  
add_subscriber.postal_code,  
offer_id,  
offer_target,  
offer_name,  
offer_opportunity_name,  
offer_opportunity_description,  
opportunity_id,  
payment_due_date,  
bank_trace_id,  
disbursement_date,  
billing_address_id,  
gross_revenue_this_disbursement_id_or_invoiced,  
gross_refund_this_disbursement_id_or_invoiced,  
cogs_this_disbursement_id_or_invoiced,  
cogs_refund_this_disbursement_id_or_invoiced,  
aws_rev_share_this_disbursement_id_or_invoiced,
```

```
aws_refund_share_this_disbursement_id_or_invoiced,  
aws_tax_share_this_disbursement_id_or_invoiced,  
aws_tax_share_listing_fee_this_disbursement_id_or_invoiced,  
aws_tax_share_refund_this_disbursement_id_or_invoiced,  
aws_tax_share_refund_listing_fee_this_disbursement_id_or_invoiced,  
seller_tax_share_this_disbursement_id_or_invoiced,  
seller_tax_share_refund_this_disbursement_id_or_invoiced,  
balance_adjustment_this_disbursement_id_or_invoiced,  
seller_rev_credit_this_disbursement_id_or_invoiced,  
aws_ref_fee_credit_this_disbursement_id_or_invoiced,  
gross_revenue_invoiced,  
gross_refund_invoiced,  
cogs_invoiced,  
cogs_refund_invoiced,  
aws_rev_share_invoiced,  
aws_refund_share_invoiced,  
aws_tax_share_invoiced,  
aws_tax_share_listing_fee_invoiced,  
aws_tax_share_refund_invoiced,  
aws_tax_share_refund_listing_fee_invoiced,  
seller_tax_share_invoiced,  
seller_tax_share_refund_invoiced,  
balance_adjustment_invoiced,  
seller_rev_credit_invoiced,  
aws_ref_fee_credit_invoiced,  
gross_revenue_disbursed,  
gross_refund_disbursed,  
cogs_disbursed,  
cogs_refund_disbursed,  
aws_rev_share_disbursed,  
aws_refund_share_disbursed,  
aws_tax_share_disbursed,  
aws_tax_share_listing_fee_disbursed,  
aws_tax_share_refund_disbursed,  
aws_tax_share_refund_listing_fee_disbursed,  
seller_tax_share_disbursed,  
seller_tax_share_refund_disbursed,  
balance_adjustment_disbursed,  
seller_rev_credit_disbursed,  
aws_ref_fee_credit_disbursed,  
last_disbursement_date,  
last_disbursement_id,  
last_disburse_bank_trace_id,  
disbursement_date_list,
```

```

    disburse_bank_trace_id_list,
    product_code,
    manufacturer_aws_account_id,
    manufacturer_account_id,
    acc_manu.mailing_company_name,
    proposer_account_id,
    acc_proposer.aws_account_id
),

billed_revenue as (
  select
    -----
    -- Invoice Info --
    -----
    buyer_invoice_date as Invoice_Date,
    Payment_Due_Date as Payment_Due_Date,
    concat(
      'Net ',
      case
        when abs(date_diff('Day', Payment_due_date, buyer_invoice_date))>180 then
'180+'
        else cast(abs(date_diff('Day', Payment_due_date, buyer_invoice_date)) as
varchar)
      end,
      ' days'
    ) as payment_terms,
    buyer_invoice_id as Invoice_ID,
    coalesce(
      --empty value in Athena shows as '', change all '' value to null
      case when seller_invoice_id = '' then null else seller_invoice_id end,
      'Not applicable') as Listing_Fee_Invoice_ID,

    -----
    --End user Information --
    -----
    coalesce(
      --empty value in Athena shows as '', change all '' value to null
      case when End_User_Company_Name = '' then null else End_User_Company_Name end,
      'Not available') as End_User_Company_Name,
    End_User_AWS_Account_ID,
    End_User_Encrypted_Account_ID,
    End_User_Email_Domain,
    End_User_City,
    End_User_State as End_User_State_or_Region,

```

```
End_User_Country,  
End_User_Postal_Code,  
End_User_Address_ID,  
  
-----  
--Subscriber Information --  
-----  
case  
  when Agreement_Id is null or Agreement_ID = '' then 'Not available'  
  when Subscriber_Company_Name is null or Subscriber_Company_Name = '' then 'Not  
provided'  
  else Subscriber_Company_Name  
  end as Subscriber_Company_Name,  
case  
  when Agreement_Id is null or Agreement_ID = '' then 'Not available'  
  else Subscriber_AWS_Account_ID  
  end as Subscriber_AWS_Account_ID,  
case  
  when Agreement_Id is null or Agreement_ID = '' then 'Not available'  
  else Subscriber_Encrypted_Account_ID  
  end as Subscriber_Encrypted_Account_ID,  
case  
  when Agreement_Id is null or Agreement_ID = '' then 'Not available'  
  when Subscriber_Email_Domain is null or Subscriber_Email_Domain = '' then 'Not  
provided'  
  else Subscriber_Email_Domain  
  end as Subscriber_Email_Domain,  
case  
  when Agreement_id is null or Agreement_ID = '' then 'Not available'  
  when Subscriber_City is null or Subscriber_City = '' then 'Not provided'  
  else Subscriber_City  
  end as Subscriber_City,  
case  
  when Agreement_Id is null or Agreement_ID = '' then 'Not available'  
  when Subscriber_State is null or Subscriber_State = '' then 'Not provided'  
  else Subscriber_State  
  end as Subscriber_State_or_Region,  
case  
  when Agreement_Id is null or Agreement_ID = '' then 'Not available'  
  when Subscriber_Country is null or Subscriber_Country = '' then 'Not provided'  
  else Subscriber_Country  
  end as Subscriber_Country,  
case  
  when Agreement_Id is null or Agreement_ID = '' then 'Not available'
```

```

    when Subscriber_Postal_Code is null or Subscriber_Postal_Code = '' then 'Not
provided'
    else Subscriber_Postal_Code
    end as Subscriber_Postal_Code,
case
    when Agreement_ID is null or Agreement_ID = '' then 'Not available'
    when Subscriber_Address_ID is null or Subscriber_Address_ID = '' then 'Not
provided'
    else Subscriber_Address_ID
    end as Subscriber_Address_ID,

-----
-- Procurement Info --
-----

-- product title at time of invoice. It is possible that the title changes over
time and therefore there may be multiple product titles mapped to a single product id.
coalesce(
    --empty value in Athena shows as '', change all '' value to null
    case when Product_Title = '' then null else Product_Title end,
    'Not provided') as Product_Title,
-- offer name at time of invoice. It is possible that the name changes over time
therefore there may be multiple offer names mapped to a single offer id.
case
    when Agreement_Id is null or Agreement_ID = '' then 'Not available'
    when (Offer_Name is null or Offer_Name = '') and Offer_Target = 'Public' then
'Not applicable'
    else Offer_Name
    end as Offer_Name,
case
    when Agreement_Id is null or Agreement_ID = ''
    then 'Not available'
    else Offer_ID
    end as Offer_ID,
-- offer target at time of invoice.,
case
    when Agreement_Id is null or Agreement_ID = '' then 'Not available'
    else Offer_Target
    end as Offer_Visibility,
coalesce(
    --empty value in Athena shows as '', change all '' value to null
    case when Agreement_ID = '' then null else Agreement_ID end,
    'Not available') as Agreement_ID,
Agreement_Start_Date,
Agreement_Acceptance_Date,

```

```

Agreement_End_Date,
Usage_Period_Start_Date,
Usage_Period_End_Date,

-----
-- Disbursement Info --
-----

case
  when Disbursement_Flag = 'Yes' then 'Disbursed'
  when Disbursement_Flag = 'No' then 'Not Disbursed'
  else 'Other'
end as Disbursement_Status,
last_disbursement_date as disbursement_date,
case
  when Disbursement_Flag = 'No' then 'Not applicable'
  when disburse_bank_trace_id_list is null or disburse_bank_trace_id_list = '' then
'Not available'
  else disburse_bank_trace_id_list
end as disburse_bank_trace_id,

-----
-- Revenues --
-----

-- We are rounding the sums using 2 decimal precision
-- Note that the rounding method might differ between SQL implementations.
-- The monthly revenue report is using RoundingMode.HALF_UP. This might create tiny
discrepancies between this SQL output
-- and the legacy report
round(gross_revenue_invoiced,2) as Gross_Revenue,
round(gross_refund_invoiced,2) as Gross_Refund,
round(aws_rev_share_invoiced,2) as Listing_Fee,
round(aws_refund_share_invoiced,2) as Listing_Fee_Refund,
truncate(
  case
    when gross_revenue_invoiced != 0 then abs(aws_rev_share_invoiced/
gross_revenue_invoiced)
    when gross_refund_invoiced != 0 then abs(aws_refund_share_invoiced/
gross_refund_invoiced)
    else 0
  end
  ,4) as Listing_Fee_Percentage,
round(seller_tax_share_invoiced,2) as Seller_Tax_Share,
round(seller_tax_share_refund_invoiced,2) as Seller_Tax_Share_Refund,
round(aws_tax_share_invoiced,2) as AWS_Tax_Share,

```

```

round(aws_tax_share_refund_invoiced,2) as AWS_Tax_Share_Refund,
round(aws_tax_share_listing_fee_invoiced,2) as AWS_Tax_Share_Listing_Fee,
round(aws_tax_share_refund_listing_fee_invoiced,2) as
AWS_Tax_Share_Refund_Listing_Fee,
round(cogs_invoiced,2) as Wholesale_cost,
round(cogs_refund_invoiced,2) as Wholesale_cost_Refund,
round(seller_net_revenue,2) as Seller_Net_Revenue,
currency as Currency,

substring(internal_buyer_line_item_id,1,strpos(internal_buyer_line_item_id,'-')-1)
as Transaction_Reference_ID,
broker_id as AWS_seller_of_record,

-----
-- Resale info --
-----
case
  when Opportunity_Id is null or Opportunity_Id = '' then
    case
      when Offer_Target = 'Public' then 'Not applicable'
      when (Offer_Target is null or Offer_Target = '') and (Agreement_Id is not
null and Agreement_Id != '') then 'Not applicable'
      else null
    end
  else Opportunity_Id
end as Resale_authorization_ID,
case
  when Offer_Opportunity_Name is null or Offer_Opportunity_Name = '' then
    case
      when Offer_Target = 'Public' then 'Not applicable'
      when (Offer_Target is null or Offer_Target = '') and (Agreement_Id is not
null and Agreement_Id != '') then 'Not applicable'
      else null
    end
  else Offer_Opportunity_Name
end as Resale_authorization_name,
case
  when Offer_Opportunity_Description is null or Offer_Opportunity_Description = ''
then
  case
    when Offer_Target = 'Public' then 'Not applicable'
    when (Offer_Target is null or Offer_Target = '') and (Agreement_Id is not null
and Agreement_Id != '') then 'Not applicable'
    else null
  end
end

```

```

        end
    else Offer_Opportunity_Name
end as Resale_authorization_description,
case
    when (Reseller_AWS_Account_ID is not null and Reseller_AWS_Account_ID != '')
        and (Reseller_Company_Name is null or Reseller_Company_Name = '') then 'Not
available'
    when (Reseller_AWS_Account_ID is null or Reseller_AWS_Account_ID = '')
        and (opportunity_id is null or opportunity_id = '') then 'Not applicable'
    when (select seller_account_id from seller_account) <>
manufacturer_aws_account_id
        and (Reseller_AWS_Account_ID is null or Reseller_AWS_Account_ID = '') then 'Not
applicable'
    else Reseller_Company_Name
end as Reseller_Company_Name,
case
    when (Reseller_AWS_Account_ID is null or Reseller_AWS_Account_ID = '')
        and (Opportunity_Id is null or Opportunity_Id = '') then 'Not applicable'
    when (select seller_account_id from seller_account) <>
manufacturer_aws_account_id
        and (Reseller_AWS_Account_ID is null or Reseller_AWS_Account_ID = '') then 'Not
applicable'
    else Reseller_AWS_Account_ID
end as Reseller_AWS_Account_ID,

-----
-- Payer Information --
-----
coalesce(
    --empty value in Athena shows as '', change all '' value to null
    case when Payer_Company_Name = '' then null else Payer_Company_Name end,
    'Not available') as Payer_Company_Name,
Payer_AWS_Account_ID,
Payer_Encrypted_Account_ID,
Payer_Email_Domain,
Payer_City,
Payer_State as Payer_State_or_Region,
Payer_Country,
Payer_Postal_Code,
Payer_Address_ID,

-----
-- ISV Information --
-----

```

```

    manufacturer_aws_account_id as ISV_Account_ID,
    coalesce(
      --empty value in Athena shows as '', change all '' value to null
      case when Manufacturer_Company_Name = '' then null else Manufacturer_Company_Name
    end,
    'Not available') as ISV_Company_Name,

    -----
    -- Products info --
    -----
    Legacy_Product_ID,
    coalesce(
      --empty value in Athena shows as '', change all '' value to null
      case when Product_ID = '' then null else Product_ID end,
      'Not provided') as Product_ID,
    Product_Code
from
  line_items_with_window_functions_enrich_offer_product_address_name as line
where disbursement_id_or_invoiced = '<invoiced>'

)

select *
from billed_revenue
where invoice_date >= date_add('DAY', -90, current_date)
--where invoice_date between cast('2023-01-01' as timestamp) and cast('2024-03-01' as
timestamp)

```

Faktur yang tidak ditagih atau dicairkan

Untuk menemukan faktur yang belum dikumpulkan atau dicairkan, Anda dapat menjalankan serangkaian kueri seperti contoh berikut. Kueri dibangun satu sama lain untuk membuat laporan Koleksi dan pencairan. Anda dapat menggunakan contoh seperti yang ditunjukkan, atau menyesuaikannya untuk data dan kasus penggunaan Anda.

Komentar dalam kueri menjelaskan apa yang dilakukan kueri, dan bagaimana memodifikasinya.

```

-- Collections and disbursements report

-- General note: When running this query, we assume that the data ingested in the
database uses

```

```

-- two time axes (the valid_from column and the update_date column).
-- See documentation for more details: https://docs.aws.amazon.com/marketplace/latest/
userguide/data-feed.html#data-feed-details

-- An account_id has several valid_from dates (each representing a separate revision of
the data)
-- but because of bi-temporality, an account_id + valid_from tuple can appear multiple
times with a different update_date.
-- We are only interested in the most recent tuple (ie, uni-temporal model)
with accounts_with_uni_temporal_data as (
  select
    account_id,
    aws_account_id,
    encrypted_account_id,
    mailing_address_id,
    tax_address_id,
    tax_legal_name,
    from_iso8601_timestamp(valid_from) as valid_from,
    tax_registration_number
  from
    (
      select
        account_id,
        aws_account_id,
        encrypted_account_id,
        mailing_address_id,
        tax_address_id,
        tax_legal_name,
        valid_from,
        delete_date,
        tax_registration_number,
        row_number() over (partition by account_id, valid_from order by
from_iso8601_timestamp(update_date) desc) as row_num
      from
        accountfeed_v1
    )
  where
    -- keep latest ...
    row_num = 1
    -- ... and remove the soft-deleted one.
    and (delete_date is null or delete_date = '')
),

accounts_with_history as (

```

```

with accounts_with_history_with_extended_valid_from as (
  select
    account_id,
    -- sometimes, this columns gets imported as a "bigint" and loses heading 0s ->
    casting to a char and re-adding heading 0s (if need be)
    substring('000000000000' || cast(aws_account_id as varchar), -12) as aws_account_id,
    encrypted_account_id,
    mailing_address_id,
    tax_address_id,
    tax_legal_name,
    -- The start time of account valid_from is extended to '1970-01-01 00:00:00',
because:
    -- ... in tax report transformations, some tax line items with invoice_date
cannot
    -- ... fall into the default valid time range of the associated account
case
  when lag(valid_from) over (partition by account_id order by valid_from asc) is
null
    then cast('1970-01-01 00:00:00' as timestamp)
    else valid_from
end as valid_from
from accounts_with_uni_temporal_data
)
select
  account_id,
  aws_account_id,
  encrypted_account_id,
  mailing_address_id,
  tax_address_id,
  tax_legal_name,
  valid_from,
  coalesce(
    lead(valid_from) over (partition by account_id order by valid_from asc),
    cast('2999-01-01 00:00:00' as timestamp)
  ) as valid_to
from
  accounts_with_history_with_extended_valid_from
),

-- An address_id has several valid_from dates (each representing a separate revision of
the data)
-- but because of bi-temporality, an account_id + valid_from tuple can appear multiple
times with a different update_date.
-- We are only interested in the most recent tuple (ie, uni-temporal model)

```

```
address_with_uni_temporal_data as (  
  select  
    from_iso8601_timestamp(valid_from) as valid_from,  
    address_id,  
    company_name,  
    email_domain,  
    country_code,  
    state_or_region,  
    city,  
    postal_code,  
    row_num  
  from  
  (  
    select  
      valid_from,  
      update_date,  
      delete_date,  
      address_id,  
      company_name,  
      email_domain,  
      country_code,  
      state_or_region,  
      city,  
      postal_code,  
      row_number() over (partition by address_id, valid_from order by  
from_iso8601_timestamp(update_date) desc) as row_num  
    from  
      addressfeed_v1  
  )  
  where  
    -- keep latest ...  
    row_num = 1  
    -- ... and remove the soft-deleted one.  
    and (delete_date is null or delete_date = '')  
),  
  
-- We are only interested in the most recent tuple (BTW: a given address is not  
  supposed to change over time but when bugs ;-) so this query mainly does nothing)  
address_with_latest_revision as (  
  select  
    valid_from,  
    address_id,  
    company_name,  
    email_domain,
```

```
country_code,
state_or_region,
city,
postal_code,
row_num_latest_revision
from
(
  select
    valid_from,
    address_id,
    company_name,
    email_domain,
    country_code,
    state_or_region,
    city,
    postal_code,
    row_number() over (partition by address_id order by valid_from desc) as
row_num_latest_revision
  from
    address_with_uni_temporal_data
)
where
  row_num_latest_revision = 1
),

accounts_with_history_with_company_name as (
  select
    awh.account_id,
    awh.aws_account_id,
    awh.encrypted_account_id,
    awh.mailing_address_id,
    awh.tax_address_id,
    coalesce(
      --empty value in Athena shows as '', change all '' value to null
      case when address.company_name = '' then null else address.company_name end,
      awh.tax_legal_name) as mailing_company_name,
    address.email_domain,
    awh.valid_from,
    -- For BYOL, the agreement might be accepted (using some external non-AWS system or
    manual process) days before
    -- that BYOL agreement is entered into AWS Marketplace by the buyer. Therefore, the
    buyer is permitted to manually
    -- enter a backdated acceptance date, which might predate the point in time when
    the account was created.
```

```

-- To work around this, we need to adjust the valid_from of the account to be
-- earlier than the earliest possible backdated BYOL agreement acceptance date.
case
  when lag(awh.valid_from) over (partition by aws_account_id order by
awh.valid_from asc) is null
    then date_add('Day', -212, awh.valid_from)
    -- 212 is the longest delay between acceptance_date of the agreement and the
account start_Date
    else awh.valid_from
  end as valid_from_adjusted,
  awh.valid_to
from accounts_with_history as awh
left join address_with_latest_revision as address on
  awh.mailing_address_id = address.address_id and awh.mailing_address_id is not null
),

-- An agreement_id has several valid_from dates (each representing an agreement
revision)
-- but because of bi-temporality, an agreement_id + valid_from tuple can appear
multiple times with a different update_date.
-- We are only interested in the most recent tuple (ie, uni-temporal model)
agreements_with_uni_temporal_data as (
  select
    agreement_id,
    origin_offer_id,
    proposer_account_id,
    acceptor_account_id,
    agreement_revision,
    from_iso8601_timestamp(valid_from) as valid_from,
    from_iso8601_timestamp(start_date) as start_date,
    from_iso8601_timestamp(end_date) as end_date,
    from_iso8601_timestamp(acceptance_date) as acceptance_date,
    agreement_type,
    previous_agreement_id,
    agreement_intent
  from
    (
      select
        --empty value in Athena shows as '', change all '' value to null
        case when agreement_id = '' then null else agreement_id end as agreement_id,
        origin_offer_id,
        proposer_account_id,
        acceptor_account_id,
        agreement_revision,

```

```

    valid_from,
    delete_date,
    start_date,
    end_date,
    acceptance_date,
    agreement_type,
    previous_agreement_id,
    agreement_intent,
    row_number() over (partition by agreement_id, valid_from order by
from_iso8601_timestamp(update_date) desc) as row_num
  from
    -- TODO change to agreementfeed_v1 when Agreement Feed is GA'ed
    agreementfeed
)
where
  -- keep latest ...
  row_num = 1
  -- ... and remove the soft-deleted one.
  and (delete_date is null or delete_date = '')
),

agreements_with_history as (
  with agreements_with_window_functions as (
    select
      agreement_id,
      origin_offer_id as offer_id,
      proposer_account_id,
      acceptor_account_id,
      agreement_revision,
      start_date,
      end_date,
      acceptance_date,
      -- The start time of agreement valid_from is extended to '1970-01-01 00:00:00',
because:
      -- ... in usage report transformations, some usage line items with usage_date
cannot
      -- ... fall into the default valid time range of the associated agreement
      case
        when lag(valid_from) over (PARTITION BY agreement_id order by valid_from asc)
is null
          then timestamp '1970-01-01 00:00:00'
          else valid_from
        end as valid_from,
      coalesce(

```

```

        lead(valid_from) over (partition by agreement_id order by valid_from asc),
        timestamp '2999-01-01 00:00:00'
    ) as valid_to,
    rank() over (partition by agreement_id order by valid_from asc) version,
    agreement_type,
    previous_agreement_id,
    agreement_intent
from
    agreements_with_uni_temporal_data
)
select
    agreement_id,
    offer_id,
    proposer_account_id,
    acceptor_account_id,
    agreement_revision,
    start_date,
    end_date,
    acceptance_date,
    valid_from,
    case
        when version=1 and valid_from < timestamp '2023-03-03 06:16:08.743' then
timestamp '1970-01-01'
        -- The following 60 minute adjustment is to handle special case where When
Renewal happens for a contract
        when version=1 then date_add('minute',-60,valid_from)
        else valid_from
    end as valid_from_adjusted,
    valid_to,
    agreement_type,
    previous_agreement_id,
    agreement_intent
from
    agreements_with_window_functions
),

-- An offer_id has several valid_from dates (each representing an offer revision)
-- but because of bi-temporality, an offer_id + valid_from tuple can appear multiple
times with a different update_date.
-- We are only interested in the most recent tuple (ie, uni-temporal model)
offers_with_uni_temporal_data as (
    select
        from_iso8601_timestamp(valid_from) as valid_from,
        from_iso8601_timestamp(update_date) as update_date,

```

```
    from_iso8601_timestamp(delete_date) as delete_date,
    offer_id,
    offer_revision,
    name,
    expiration_date,
    opportunity_id,
    opportunity_name,
    opportunity_description,
    seller_account_id
from
(
  select
    valid_from,
    update_date,
    delete_date,
    offer_id,
    offer_revision,
    name,
    expiration_date,
    opportunity_id,
    opportunity_name,
    opportunity_description,
    seller_account_id,
    row_number() over (partition by offer_id, valid_from order by
from_iso8601_timestamp(update_date) desc) as row_num
  from
    offerfeed_v1
)
where
  -- keep latest ...
  row_num = 1
  -- ... and remove the soft-deleted one.
  and (delete_date is null or delete_date = '')
),

-- Here, we build the validity time range (adding valid_to on top of valid_from) of
  each offer revision.
-- We will use it to get Offer name at invoice time.
-- NB: If you'd rather get "current" offer name, un-comment
  "offers_with_latest_revision"
offers_with_history as (
  select
    offer_id,
    offer_revision,
```

```

name,
opportunity_id,
opportunity_name,
opportunity_description,
valid_from,
-- When we try to look up an offer revision as at the acceptance date of a BYOL
agreement, we run into a problem.
-- For BYOL, the agreement might be accepted (using some external non-AWS system or
manual process) days before
-- that BYOL agreement is entered into AWS Marketplace by the buyer. Therefore, the
buyer is permitted to manually
-- enter a backdated acceptance date, which might predate the point in time when
the first revision of the offer
-- was created. To work around this, we need to adjust the valid_from on the first
revision of the offer to be
-- earlier than the earliest possible backdated BYOL agreement acceptance date.
case
  when lag(valid_from) over (partition by offer_id order by valid_from asc) is null
and valid_from < cast('2021-04-01' as timestamp)
  then date_add('Day', -3857, valid_from)
  -- 3857 is the longest delay between acceptance_date of an agreement and the
first revision of the offer
  when lag(valid_from) over (partition by offer_id order by valid_from asc) is null
and valid_from >= cast('2021-04-01' as timestamp)
  then date_add('Day', -1460, valid_from)
  --after 2021 for the two offers we need to adjust for 2 more years
  else valid_from
end as valid_from_adjusted,
coalesce(
  lead(valid_from) over (partition by offer_id order by valid_from asc),
  cast('2999-01-01 00:00:00' as timestamp))
as valid_to
from offers_with_uni_temporal_data
),
-- provided for reference only if you are interested into get "current" offer name
-- (ie. not used afterwards)
offers_with_latest_revision as (
  select
    offer_id,
    offer_revision,
    name,
    opportunity_name,
    opportunity_description,
    valid_from,

```

```
    null valid_to
from
(
  select
    offer_id,
    offer_revision,
    name,
    opportunity_name,
    opportunity_description,
    valid_from,
    null valid_to,
    row_number() over (partition by offer_id order by valid_from desc) as
row_num_latest_revision
  from
    offers_with_uni_temporal_data
)
where
  row_num_latest_revision = 1
),

-- An offer_target_id has several valid_from dates (each representing an offer
revision)
-- but because of bi-temporality, an offer_target_id + valid_from tuple can appear
multiple times with a different update_date.
-- We are only interested in the most recent tuple (ie, uni-temporal model)
offer_targets_with_uni_temporal_data as (
  select
    from_iso8601_timestamp(valid_from) as valid_from,
    from_iso8601_timestamp(update_date) as update_date,
    from_iso8601_timestamp(delete_date) as delete_date,
    offer_target_id,
    offer_id,
    offer_revision,
    target_type,
    polarity,
    value
  from
  (
    select
      valid_from,
      update_date,
      delete_date,
      offer_target_id,
      offer_id,
```

```

    offer_revision,
    target_type,
    polarity,
    value,
    row_number() over (partition by offer_target_id, valid_from order by
from_iso8601_timestamp(update_date) desc) as row_num
  from
    offertargetfeed_v1
)
where
  -- keep latest ...
  row_num = 1
  -- ... and remove the soft-deleted one.
  and (delete_date is null or delete_date = '')
),

offer_target_type as (
  select
    offer_id,
    offer_revision,
    substring(
      -- The first character indicates the priority (lower value means higher
precedence):
      min(
        case
          when offer_target.target_type='BuyerAccounts' then '1Private'
          when offer_target.target_type='ParticipatingPrograms' then '2Program:'||
cast(offer_target.value as varchar)
          when offer_target.target_type='CountryCodes' then '3GeoTargeted'
          -- well, there is no other case today, but rather be safe...
          else '4Other Targeting'
        end
      ),
      -- Remove the first character that was only used for th priority in the "min"
aggregate function:
      2
    ) as offer_target
  from
    offer_targets_with_uni_temporal_data as offer_target
  group by
    offer_id,
    offer_revision
),

```

```
offers_with_history_with_target_type as (  
  select  
    offer.offer_id,  
    offer.offer_revision,  
    -- even though today it is not possible to combine several types of targeting in a  
    single offer, let's ensure the query is still predictable if this gets possible in the  
    future  
    max(  
      case  
        when off_tgt.offer_target is null then 'Public'  
        else off_tgt.offer_target  
      end  
    ) as offer_target,  
    min(offer.name) as name,  
    min(offer.opportunity_name) as opportunity_name,  
    min(offer.opportunity_description) as opportunity_description,  
    offer.valid_from,  
    offer.valid_from_adjusted,  
    offer.valid_to,  
    offer.opportunity_id  
  from  
    offers_with_history as offer  
  left join offer_target_type as off_tgt on  
    offer.offer_id = off_tgt.offer_id  
    and offer.offer_revision = off_tgt.offer_revision  
  group by  
    offer.offer_id,  
    offer.offer_revision,  
    offer.valid_from,  
    offer.valid_from_adjusted,  
    offer.valid_to,  
    offer.opportunity_id  
),  
  
-- provided for reference only if you are interested into get "current" offer targets  
-- (ie. not used afterwards)  
offers_with_latest_revision_with_target_type as (  
  select  
    offer.offer_id,  
    offer.offer_revision,  
    -- even though today it is not possible to combine several types of targeting in a  
    single offer, let's ensure the query is still predictable if this gets possible in the  
    future  
    max(  
      case  
        when off_tgt.offer_target is null then 'Public'  
        else off_tgt.offer_target  
      end  
    ) as offer_target,  
    min(offer.name) as name,  
    min(offer.opportunity_name) as opportunity_name,  
    min(offer.opportunity_description) as opportunity_description,  
    offer.valid_from,  
    offer.valid_from_adjusted,  
    offer.valid_to,  
    offer.opportunity_id  
  from  
    offers_with_history as offer  
  left join offer_target_type as off_tgt on  
    offer.offer_id = off_tgt.offer_id  
    and offer.offer_revision = off_tgt.offer_revision  
  group by  
    offer.offer_id,  
    offer.offer_revision,  
    offer.valid_from,  
    offer.valid_from_adjusted,  
    offer.valid_to,  
    offer.opportunity_id  
),
```

```

distinct
case
  when off_tgt.target_type is null then 'Public'
  when off_tgt.target_type='BuyerAccounts' then 'Private'
  when off_tgt.target_type='ParticipatingPrograms' then 'Program:'||
cast(off_tgt.value as varchar)
  when off_tgt.target_type='CountryCodes' then 'GeoTargeted'
  -- well, there is no other case today, but rather be safe...
  else 'Other Targeting'
end
) as offer_target,
min(offer.name) as name,
min(offer.opportunity_name) as opportunity_name,
min(offer.opportunity_description) as opportunity_description,
offer.valid_from,
offer.valid_to
from
  offers_with_latest_revision offer
-- left joining because public offers don't have targets
left join offer_targets_with_uni_temporal_data off_tgt on
  offer.offer_id=off_tgt.offer_id and offer.offer_revision=off_tgt.offer_revision
group by
  offer.offer_id,
  offer.offer_revision,
  -- redundant with offer_revision, as each revision has a dedicated valid_from (but
  cleaner in the group by)
  offer.valid_from,
  offer.valid_to
),

-- A product_id has several valid_from dates (each representing a product revision),
-- but because of bi-temporality, each product_id + valid_from tuple can appear
multiple times with a different update_date.
-- We are only interested in the most recent tuple (ie, uni-temporal model)
products_with_uni_temporal_data as (
  select
    from_iso8601_timestamp(valid_from) as valid_from,
    from_iso8601_timestamp(update_date) as update_date,
    from_iso8601_timestamp(delete_date) as delete_date,
    product_id,
    manufacturer_account_id,
    product_code,
    title
  from

```

```

(
  select
    valid_from,
    update_date,
    delete_date,
    product_id,
    manufacturer_account_id,
    product_code,
    title,
    row_number() over (partition by product_id, valid_from order by
from_iso8601_timestamp(update_date) desc) as row_num
  from
    productfeed_v1
)
where
  -- keep latest ...
  row_num = 1
  -- ... and remove the soft-deleted one.
  and (delete_date is null or delete_date = '')
),

products_with_history as (
  select
    product_id,
    title,
    valid_from,
    -- Offerv2 can have upto 50 years and Offerv3 is upto 5 years of past date
    case
      when lag(valid_from) over (partition by product_id order by valid_from asc) is
null and valid_from < cast('2021-04-01' as timestamp)
        then date_add('Day', -3857, valid_from)
        -- 3827 is the longest delay between acceptance_date of an agreement and the
product
        -- we are keeping 3857 as a consistency between the offers and products
      when lag(valid_from) over (partition by product_id order by valid_from asc) is
null and valid_from >= cast('2021-04-01' as timestamp)
        then date_add('Day', -2190, valid_from)
        --after 2021 for the two offers we need to adjust for 2 more years
      else valid_from
    end as valid_from_adjusted,
    coalesce(
      lead(valid_from) over (partition by product_id order by valid_from asc),
      cast('2999-01-01 00:00:00' as timestamp)
    ) as valid_to,

```

```
    product_code,  
    manufacturer_account_id  
from  
    products_with_uni_temporal_data  
) ,
```

```
legacy_products as (  
    select  
        legacy_id,  
        new_id  
    from  
        legacyidmappingfeed_v1  
    where  
        mapping_type='PRODUCT'  
    group by  
        legacy_id,  
        new_id  
) ,
```

```
-- A given billing_event_id represents an accounting event and thus has only one  
valid_from date,  
-- but because of bi-temporality, a billing_event_id (+ its valid_from) can appear  
multiple times with a different update_date.  
-- We are only interested in the most recent tuple (ie, uni-temporal model)
```

```
billing_events_with_uni_temporal_data as (  
    select  
        billing_event_id,  
        valid_from,  
        update_date,  
        delete_date,  
        invoice_date,  
        transaction_type,  
        transaction_reference_id,  
        parent_billing_event_id,  
        bank_trace_id,  
        broker_id,  
        product_id,  
        disbursement_billing_event_id,  
        action,  
        from_account_id,  
        to_account_id,  
        end_user_account_id,  
        billing_address_id,  
        amount,
```

```
currency,
balance_impacting,
--empty value in Athena shows as '', change all '' value to null
case when agreement_id = '' then null else agreement_id end as agreement_id,
invoice_id,
payment_due_date,
usage_period_start_date,
usage_period_end_date,
buyer_transaction_reference_id,
row_num
from
(
select
  billing_event_id,
  from_iso8601_timestamp(valid_from) as valid_from,
  from_iso8601_timestamp(update_date) as update_date,
  delete_date,
  from_iso8601_timestamp(invoice_date) as invoice_date,
  transaction_type,
  transaction_reference_id,
  parent_billing_event_id,
  -- casting in case data was imported as number
  cast(bank_trace_id as varchar) as bank_trace_id,
  broker_id,
  product_id,
  disbursement_billing_event_id,
  action,
  from_account_id,
  to_account_id,
  end_user_account_id,
  billing_address_id,
  -- casting in case data was imported as varchar
  cast(amount as decimal(38,6)) as amount,
  currency,
  balance_impacting,
  agreement_id,
  invoice_id,
  case when payment_due_date is null or payment_due_date = '' then null else
from_iso8601_timestamp(payment_due_date) end as payment_due_date,
  from_iso8601_timestamp(usage_period_start_date) as usage_period_start_date,
  from_iso8601_timestamp(usage_period_end_date) as usage_period_end_date,
  buyer_transaction_reference_id,
  row_number() over (partition by billing_event_id, valid_from order by
from_iso8601_timestamp(update_date) desc) as row_num
```

```

    from
      billingeventfeed_v1
  )
where
  -- keep latest ...
  row_num = 1
  -- ... and remove the soft-deleted one.
  and (delete_date is null or delete_date = '')
),

-- Here we select the account_id of the current seller (We identify this by looking for
the to_account_id related to revenue transactions).
-- We will use it later to distinguish own agreements from agreements generated by
channel partners.
seller_account as (
  select
    from_account_id as seller_account_id
  from
    billing_events_with_uni_temporal_data bill
  where
    -- Assumption here is only seller will pay listing fee. As of 12/21/2021, there are
cases that Channel partner have 0 listing fee for CPP0, so the amount could be 0.
    bill.transaction_type like 'AWS_REV_SHARE' and amount <= 0 and action = 'INVOICED'
  group by
    -- from_account_id is always the same for all those "listing fee" transactions ==
the seller of record himself.
    -- If this view returns more than 1 record, the overall query will fail (on
purpose). Please contact AWS Marketplace if this happens.
    from_account_id
),

billing_event_with_business_flags as (
  select
    bl.billing_event_id,
    bl.end_user_account_id,
    bl.agreement_id,
    aggrement.proposer_account_id,
    aggrement.offer_id,
    aggrement.acceptor_account_id,
  case
    -- For AWS and BALANCE_ADJUSTMENT, the billing event feed will show the "AWS
Marketplace" account as the
    -- receiver of the funds and the seller as the payer. We are not interested in
this information here.

```

```

-- Null values will be ignored by the `max` aggregation function.
when bl.transaction_type like 'AWS%' then null
-- For BALANCE_ADJUSTMENT, payer is seller themselves
when bl.invoice_id is null then bl.to_account_id
-- We get the payer of the invoice from *any* transaction type that is not AWS
and not BALANCE_ADJUSTMENT (because they are the same for a given end user + agreement
+ product).
    else bl.from_account_id
end as payer_account_id,
bl.product_id,
bl.action,
bl.transaction_type,
bl.parent_billing_event_id,
bl.disbursement_billing_event_id,
bl.amount,
bl.currency,
bl.balance_impacting,
bl.invoice_date,
bl.payment_due_date,
bl.usage_period_start_date,
bl.usage_period_end_date,
bl.invoice_id,
bl.billing_address_id,
bl.transaction_reference_id,
bl.buyer_transaction_reference_id,
case when disbursement.bank_trace_id = 'EMEA_MP_TEST_TRACE_ID' then null else
disbursement.bank_trace_id end as bank_trace_id,
case when disbursement.bank_trace_id = 'EMEA_MP_TEST_TRACE_ID' then null else
disbursement.invoice_date end as disbursement_date,
disbursement.billing_event_id as disbursement_id,
-- We will use disbursement_id_or_invoiced as part of the PK, so it cannot be null:
coalesce(
    --empty value in Athena shows as '', change all '' value to null
    case when disbursement.billing_event_id = '' then null else
disbursement.billing_event_id end,
    '<invoiced>') as disbursement_id_or_invoiced,
bl.broker_id,
case
    when bl.invoice_id is null /* transaction_type = 'BALANCE_ADJUSTMENT' */
    then (select seller_account_id from seller_account) || ':' ||
cast(bl.invoice_date as varchar)
    else bl.buyer_transaction_reference_id
    || '-' || case when bl.agreement_id is null or bl.agreement_id = '' then ' ' else
bl.agreement_id end

```

```

        ||'-'|| case when bl.end_user_account_id is null or bl.end_user_account_id = ''
then ' ' else bl.end_user_account_id end
        ||'-'|| coalesce(cast(bl.usage_period_start_date as varchar),' ')
        ||'-'|| coalesce(cast(bl.usage_period_end_date as varchar),' ')
    end as internal_buyer_line_item_id,
    bl.buyer_transaction_reference_id <> bl.transaction_reference_id as
is_seller_invoice,
    case when bl.transaction_type = 'SELLER_REV_SHARE' and (select seller_account_id
from seller_account) <> bl.to_account_id then true else false end as is_cog,
    case when bl.transaction_type in('SELLER_REV_SHARE_CREDIT',
'SELLER_REV_SHARE_REFUND') and (select seller_account_id from seller_account) <>
bl.to_account_id then true else false end as is_cog_refund,
    --TODO: replace below logic once we can create a logic the identify reseller/
manufacturer without depending on agreement feed
    case when aggrement.proposer_account_id <> (select seller_account_id from
seller_account) then true else false end as is_manufacturer_view_of_reseller
from
    billing_events_with_uni_temporal_data as bl
left join billing_events_with_uni_temporal_data as disbursement on
    disbursement.transaction_type like 'DISBURSEMENT%'
    and disbursement.action = 'DISBURSED'
    and disbursement.transaction_type IN ('DISBURSEMENT', 'DISBURSEMENT_FAILURE')
    and bl.disbursement_billing_event_id = disbursement.billing_event_id
left join agreements_with_history as aggrement on
    bl.agreement_id = aggrement.agreement_id
    and bl.invoice_date >= aggrement.valid_from_adjusted
    and bl.invoice_date < aggrement.valid_to
left join accounts_with_history awh on
    bl.to_account_id = awh.account_id
    and bl.invoice_date >= awh.valid_from
    and bl.invoice_date < awh.valid_to
where
    bl.transaction_type not like 'DISBURSEMENT%' and
    (bl.agreement_id is null or bl.agreement_id = ''
    or aggrement.agreement_id is not null)
),

-- listagg function in athena does not support partitioning, grouping here and then
joining to the main query
seller_invoice_list as (
    select
        internal_buyer_line_item_id,

```

```

    listagg(case when not is_seller_invoice then null else invoice_id end,',' ) within
group (order by case when not is_seller_invoice then null else cast(invoice_date as
varchar) end) as seller_invoice_id_or_null,
    listagg(case when not is_seller_invoice then null else cast(invoice_date as
varchar) end,',' ) within group (order by case when not is_seller_invoice then null
else cast(invoice_date as varchar) end) as seller_invoice_date_or_null
from
(
    -- listagg function in athena does not support ordering by another field when
distinct is used,
    -- here we first select distinct invoices and then do the listagg order by
invoice_date
    select distinct internal_buyer_line_item_id, is_seller_invoice, invoice_id,
invoice_date
    from billing_event_with_business_flags) distinct_invoices
group by internal_buyer_line_item_id
order by internal_buyer_line_item_id
),

billing_event_with_categorized_transaction as (
-- Use the flags that were created in the previous transformation in more calculated
columns:
-- NOTE: This transformation has no joins and no window functions
select
    billing_event_id,
    end_user_account_id,
    agreement_id,
    proposer_account_id,
    offer_id,
    acceptor_account_id,
    case when is_cog or is_cog_refund then null else payer_account_id end as
payer_account_id,
    product_id,
    action,
    transaction_type,
    parent_billing_event_id,
    disbursement_billing_event_id,
    amount,
    currency,
    balance_impacting,
    invoice_date,
    payment_due_date,
    usage_period_start_date,
    usage_period_end_date,

```

```

invoice_id,
billing_address_id,
transaction_reference_id,
buyer_transaction_reference_id,
bank_trace_id,
disbursement_date,
disbursement_id,
disbursement_id_or_invoiced,
broker_id,
bl.internal_buyer_line_item_id,
is_seller_invoice,
is_cog,
is_cog_refund,
is_manufacturer_view_of_reseller,

-- Buyer/seller columns:
case when is_seller_invoice then null else invoice_id end as
buyer_invoice_id_or_null,
seller_invoices.seller_invoice_id_or_null,
case when is_seller_invoice then null else invoice_date end as
buyer_invoice_date_or_null,
seller_invoices.seller_invoice_date_or_null,

-- Categorized amounts by transaction type:
case when transaction_type = 'SELLER_REV_SHARE' and not is_cog then amount else 0
end as gross_revenue,
case when transaction_type in ('SELLER_REV_SHARE_REFUND', 'SELLER_REV_SHARE_CREDIT')
and not is_cog_refund then amount else 0 end as gross_refund,
case when transaction_type = 'SELLER_REV_SHARE' and is_cog then amount else 0
end as cogs,
case when transaction_type in ('SELLER_REV_SHARE_REFUND', 'SELLER_REV_SHARE_CREDIT')
and is_cog_refund then amount else 0 end as cogs_refund,
case when transaction_type = 'AWS_REV_SHARE' then amount else 0 end as
aws_rev_share,
case when transaction_type in ('AWS_REV_SHARE_REFUND', 'AWS_REV_SHARE_CREDIT') then
amount else 0 end as aws_refund_share,
case when transaction_type = 'AWS_TAX_SHARE' and not is_seller_invoice then
amount else 0 end as aws_tax_share, -- AWS tax share from_buyer_ invoice
case when transaction_type = 'AWS_TAX_SHARE' and is_seller_invoice then
amount else 0 end as aws_tax_share_listing_fee, -- AWS tax share from_seller_ invoice
case when transaction_type = 'AWS_TAX_SHARE_REFUND' and not is_seller_invoice
then amount else 0 end as aws_tax_share_refund,
case when transaction_type = 'AWS_TAX_SHARE_REFUND' and is_seller_invoice
then amount else 0 end as aws_tax_share_refund_listing_fee,

```

```
    case when transaction_type = 'SELLER_TAX_SHARE' then amount else 0 end as
seller_tax_share,
    case when transaction_type = 'SELLER_TAX_SHARE_REFUND' then amount else 0 end as
seller_tax_share_refund,
    case when transaction_type = 'BALANCE_ADJUSTMENT' then amount else 0 end as
balance_adjustment,
    case when transaction_type = 'SELLER_REV_SHARE_CREDIT' then amount else 0 end as
seller_rev_credit,
    case when transaction_type = 'AWS_REV_SHARE_CREDIT' then amount else 0 end as
aws_ref_fee_credit
from
    billing_event_with_business_flags as bl
    left join seller_invoice_list as seller_invoices
        on bl.internal_buyer_line_item_id = seller_invoices.internal_buyer_line_item_id
),

line_items_aggregated as (
-- This transformation has the only "group by" in all of these transformations.
-- NOTE: This transformation has no joins and no window functions
select
    internal_buyer_line_item_id,
    disbursement_id,
    disbursement_id_or_invoiced,
    product_id,
    broker_id,
    currency,
    agreement_id,
    proposer_account_id,
    acceptor_account_id,
    max(payer_account_id) as payer_account_id,
    offer_id,
    end_user_account_id,
    usage_period_start_date,
    usage_period_end_date,
    max(payment_due_date) payment_due_date,
    buyer_transaction_reference_id,
    bank_trace_id,
    disbursement_date,
    max(billing_address_id) as billing_address_id,

    -- Buyer/seller columns:
    max(buyer_invoice_id_or_null) as buyer_invoice_id,
    max(seller_invoice_id_or_null) as seller_invoice_id,
    max(buyer_invoice_date_or_null) as buyer_invoice_date,
```

```

max(seller_invoice_date_or_null) as seller_invoice_date,

-- Categorized amounts by transaction type:
-- When disbursement_id_or_invoiced = '<invoiced>',    these are invoiced amounts
-- When disbursement_id_or_invoiced <> ''<invoiced>' these are disbursed amounts
for _this_ specific disbursement_id
sum(gross_revenue) as gross_revenue_this_disbursement_id_or_invoiced,
sum(gross_refund) as gross_refund_this_disbursement_id_or_invoiced,
sum(cogs) as cogs_this_disbursement_id_or_invoiced,
sum(cogs_refund) as cogs_refund_this_disbursement_id_or_invoiced,
sum(aws_rev_share) as aws_rev_share_this_disbursement_id_or_invoiced,
sum(aws_refund_share) as aws_refund_share_this_disbursement_id_or_invoiced,
sum(aws_tax_share) as aws_tax_share_this_disbursement_id_or_invoiced,
sum(aws_tax_share_listing_fee) as
aws_tax_share_listing_fee_this_disbursement_id_or_invoiced,
sum(aws_tax_share_refund) as aws_tax_share_refund_this_disbursement_id_or_invoiced,
sum(aws_tax_share_refund_listing_fee) as
aws_tax_share_refund_listing_fee_this_disbursement_id_or_invoiced,
sum(seller_tax_share) as seller_tax_share_this_disbursement_id_or_invoiced,
sum(seller_tax_share_refund) as
seller_tax_share_refund_this_disbursement_id_or_invoiced,
sum(balance_adjustment) as balance_adjustment_this_disbursement_id_or_invoiced,
sum(seller_rev_credit) as seller_rev_credit_this_disbursement_id_or_invoiced,
sum(aws_ref_fee_credit) as aws_ref_fee_credit_this_disbursement_id_or_invoiced
from
  billing_event_with_categorized_transaction as billing_categorized
group by
  internal_buyer_line_item_id,
  disbursement_id,
  disbursement_id_or_invoiced,
  broker_id,
  -- The following columns are included the in group by but they are intentionally
  omitted from the PK.
  -- These columns should have the _same_ values for each record in the PK.
  product_id,
  currency,
  agreement_id,
  proposer_account_id,
  acceptor_account_id,
  offer_id,
  end_user_account_id,
  usage_period_start_date,
  usage_period_end_date,
  buyer_transaction_reference_id,

```

```
    bank_trace_id,
    disbursement_date
),

-- listagg function in athena does not support partitioning, grouping here and then
  joining to the main query
disbursement_list as (
  select
    internal_buyer_line_item_id,
    listagg(cast(disbursement_date as varchar),',') within group (order by
cast(disbursement_date as varchar)) as disbursement_date_list,
    listagg(bank_trace_id,',') within group (order by cast(disbursement_date as
varchar)) as disburse_bank_trace_id_list
  from (
    -- listagg function in athena does not support ordering by another field when
distinct is used,
    -- here we first select distinct bank_trace_ids and then do the listagg order by
disbursement_date
    select distinct internal_buyer_line_item_id, disbursement_date, bank_trace_id
  from billing_event_with_business_flags) distinct_disbursements
  group by internal_buyer_line_item_id
  order by internal_buyer_line_item_id
),

line_items_with_window_functions as (
--add flag next step compare gross_revenue and gross_revenue_disbursed or gross_refund
and gross_refund_disbursed
  select
    line_item.internal_buyer_line_item_id,
    disbursement_id,
    disbursement_id_or_invoiced,
    product_id,
    broker_id,
    currency,
    agreement_id,
    proposer_account_id,
    acceptor_account_id,
    -- when there's aws_rev_Share adjustment/refund to a seller_rev_share invoice, it
can happen that for the same aws_rev_share invoice_id, there are multiple disbursement
events,
    -- using windows function to map payer_account_id of seller_rev_share to all
corresponding aws_rev_Share
    max(payer_account_id) over (partition by line_item.internal_buyer_line_item_id) as
payer_account_id,
```

```
offer_id,
end_user_account_id,
usage_period_start_date,
usage_period_end_date,
payment_due_date,
bank_trace_id,
disbursement_date,
billing_address_id,

-- Buyer/seller columns:
max(buyer_invoice_id) over (partition by line_item.internal_buyer_line_item_id) as
buyer_invoice_id,
seller_invoice_id,
max(buyer_invoice_date) over (partition by line_item.internal_buyer_line_item_id)
as buyer_invoice_date,
seller_invoice_date,

-- When disbursement_id_or_invoiced = '<invoiced>', these are actually invoiced
amounts
-- When disbursement_id_or_invoiced <> '<invoiced>' these are disbursed amounts
for _this_ specific disbursement_id
gross_revenue_this_disbursement_id_or_invoiced,
gross_refund_this_disbursement_id_or_invoiced,
cogs_this_disbursement_id_or_invoiced,
cogs_refund_this_disbursement_id_or_invoiced,
aws_rev_share_this_disbursement_id_or_invoiced,
aws_refund_share_this_disbursement_id_or_invoiced,
aws_tax_share_this_disbursement_id_or_invoiced,
aws_tax_share_listing_fee_this_disbursement_id_or_invoiced,
aws_tax_share_refund_this_disbursement_id_or_invoiced,
aws_tax_share_refund_listing_fee_this_disbursement_id_or_invoiced,
seller_tax_share_this_disbursement_id_or_invoiced,
seller_tax_share_refund_this_disbursement_id_or_invoiced,
balance_adjustment_this_disbursement_id_or_invoiced,
seller_rev_credit_this_disbursement_id_or_invoiced,
aws_ref_fee_credit_this_disbursement_id_or_invoiced,
-- IMPORTANT: All window functions partitioned by internal_buyer_line_item_id:

-- Invoiced amounts, categorized by transaction type:
sum(case when disbursement_id_or_invoiced = '<invoiced>' then
gross_revenue_this_disbursement_id_or_invoiced else cast(0 as decimal(38,6)) end)over
(partition by line_item.internal_buyer_line_item_id) as gross_revenue_invoiced,
```

```
sum(case when disbursement_id_or_invoiced = '<invoiced>' then
gross_refund_this_disbursement_id_or_invoiced else cast(0 as decimal(38,6)) end) over
(partition by line_item.internal_buyer_line_item_id) as gross_refund_invoiced,
sum(case when disbursement_id_or_invoiced = '<invoiced>' then
cogs_this_disbursement_id_or_invoiced else cast(0 as decimal(38,6)) end) over
(partition by line_item.internal_buyer_line_item_id) as cogs_invoiced,
sum(case when disbursement_id_or_invoiced = '<invoiced>' then
cogs_refund_this_disbursement_id_or_invoiced else cast(0 as decimal(38,6)) end) over
(partition by line_item.internal_buyer_line_item_id) as cogs_refund_invoiced,
sum(case when disbursement_id_or_invoiced = '<invoiced>' then
aws_rev_share_this_disbursement_id_or_invoiced else cast(0 as decimal(38,6)) end) over
(partition by line_item.internal_buyer_line_item_id) as aws_rev_share_invoiced,
sum(case when disbursement_id_or_invoiced = '<invoiced>' then
aws_refund_share_this_disbursement_id_or_invoiced else cast(0 as decimal(38,6))
end) over (partition by line_item.internal_buyer_line_item_id) as
aws_refund_share_invoiced,
sum(case when disbursement_id_or_invoiced = '<invoiced>' then
aws_tax_share_this_disbursement_id_or_invoiced else cast(0 as decimal(38,6)) end) over
(partition by line_item.internal_buyer_line_item_id) as aws_tax_share_invoiced,
sum(case when disbursement_id_or_invoiced = '<invoiced>' then
aws_tax_share_listing_fee_this_disbursement_id_or_invoiced else cast(0 as
decimal(38,6)) end) over (partition by line_item.internal_buyer_line_item_id) as
aws_tax_share_listing_fee_invoiced,
sum(case when disbursement_id_or_invoiced = '<invoiced>' then
aws_tax_share_refund_this_disbursement_id_or_invoiced else cast(0 as
decimal(38,6)) end) over (partition by line_item.internal_buyer_line_item_id) as
aws_tax_share_refund_invoiced,
sum(case when disbursement_id_or_invoiced = '<invoiced>' then
aws_tax_share_refund_listing_fee_this_disbursement_id_or_invoiced else cast(0 as
decimal(38,6)) end) over (partition by line_item.internal_buyer_line_item_id) as
aws_tax_share_refund_listing_fee_invoiced,
sum(case when disbursement_id_or_invoiced = '<invoiced>' then
seller_tax_share_this_disbursement_id_or_invoiced else cast(0 as decimal(38,6))
end) over (partition by line_item.internal_buyer_line_item_id) as
seller_tax_share_invoiced,
sum(case when disbursement_id_or_invoiced = '<invoiced>' then
seller_tax_share_refund_this_disbursement_id_or_invoiced else cast(0 as
decimal(38,6)) end) over (partition by line_item.internal_buyer_line_item_id) as
seller_tax_share_refund_invoiced,
sum(case when disbursement_id_or_invoiced = '<invoiced>' then
balance_adjustment_this_disbursement_id_or_invoiced else cast(0 as
decimal(38,6)) end) over (partition by line_item.internal_buyer_line_item_id) as
balance_adjustment_invoiced,
```

```

sum(case when disbursement_id_or_invoiced = '<invoiced>' then
seller_rev_credit_this_disbursement_id_or_invoiced else cast(0 as
decimal(38,6)) end) over (partition by line_item.internal_buyer_line_item_id) as
seller_rev_credit_invoiced,
sum(case when disbursement_id_or_invoiced = '<invoiced>' then
aws_ref_fee_credit_this_disbursement_id_or_invoiced else cast(0 as
decimal(38,6)) end) over (partition by line_item.internal_buyer_line_item_id) as
aws_ref_fee_credit_invoiced,

-- Total disbursed amounts (for all disbursement_id values), categorized by
transaction type:
sum(case when disbursement_id_or_invoiced '<>' '<invoiced>' then
gross_revenue_this_disbursement_id_or_invoiced else cast(0 as decimal(38,6)) end) over
(partition by line_item.internal_buyer_line_item_id) as gross_revenue_disbursed,
sum(case when disbursement_id_or_invoiced '<>' '<invoiced>' then
gross_refund_this_disbursement_id_or_invoiced else cast(0 as decimal(38,6)) end) over
(partition by line_item.internal_buyer_line_item_id) as gross_refund_disbursed,
sum(case when disbursement_id_or_invoiced '<>' '<invoiced>' then
cogs_this_disbursement_id_or_invoiced else cast(0 as decimal(38,6)) end) over
(partition by line_item.internal_buyer_line_item_id) as cogs_disbursed,
sum(case when disbursement_id_or_invoiced '<>' '<invoiced>' then
cogs_refund_this_disbursement_id_or_invoiced else cast(0 as decimal(38,6)) end) over
(partition by line_item.internal_buyer_line_item_id) as cogs_refund_disbursed,
sum(case when disbursement_id_or_invoiced '<>' '<invoiced>' then
aws_rev_share_this_disbursement_id_or_invoiced else cast(0 as decimal(38,6)) end) over
(partition by line_item.internal_buyer_line_item_id) as aws_rev_share_disbursed,
sum(case when disbursement_id_or_invoiced '<>' '<invoiced>' then
aws_refund_share_this_disbursement_id_or_invoiced else cast(0 as decimal(38,6))
end) over (partition by line_item.internal_buyer_line_item_id) as
aws_refund_share_disbursed,
sum(case when disbursement_id_or_invoiced '<>' '<invoiced>' then
aws_tax_share_this_disbursement_id_or_invoiced else cast(0 as decimal(38,6)) end) over
(partition by line_item.internal_buyer_line_item_id) as aws_tax_share_disbursed,
sum(case when disbursement_id_or_invoiced '<>' '<invoiced>' then
aws_tax_share_listing_fee_this_disbursement_id_or_invoiced else cast(0 as
decimal(38,6)) end) over (partition by line_item.internal_buyer_line_item_id) as
aws_tax_share_listing_fee_disbursed,
sum(case when disbursement_id_or_invoiced '<>' '<invoiced>' then
aws_tax_share_refund_this_disbursement_id_or_invoiced else cast(0 as
decimal(38,6)) end) over (partition by line_item.internal_buyer_line_item_id) as
aws_tax_share_refund_disbursed,
sum(case when disbursement_id_or_invoiced '<>' '<invoiced>' then
aws_tax_share_refund_listing_fee_this_disbursement_id_or_invoiced else cast(0 as

```

```

decimal(38,6)) end) over (partition by line_item.internal_buyer_line_item_id) as
aws_tax_share_refund_listing_fee_disbursed,
    sum(case when disbursement_id_or_invoiced '<>' '<invoiced>' then
seller_tax_share_this_disbursement_id_or_invoiced else cast(0 as decimal(38,6))
end) over (partition by line_item.internal_buyer_line_item_id) as
seller_tax_share_disbursed,
    sum(case when disbursement_id_or_invoiced '<>' '<invoiced>' then
seller_tax_share_refund_this_disbursement_id_or_invoiced else cast(0 as
decimal(38,6)) end) over (partition by line_item.internal_buyer_line_item_id) as
seller_tax_share_refund_disbursed,
    sum(case when disbursement_id_or_invoiced '<>' '<invoiced>' then
balance_adjustment_this_disbursement_id_or_invoiced else cast(0 as
decimal(38,6)) end) over (partition by line_item.internal_buyer_line_item_id) as
balance_adjustment_disbursed,
    sum(case when disbursement_id_or_invoiced '<>' '<invoiced>' then
seller_rev_credit_this_disbursement_id_or_invoiced else cast(0 as
decimal(38,6)) end) over (partition by line_item.internal_buyer_line_item_id) as
seller_rev_credit_disbursed,
    sum(case when disbursement_id_or_invoiced '<>' '<invoiced>' then
aws_ref_fee_credit_this_disbursement_id_or_invoiced else cast(0 as
decimal(38,6)) end) over (partition by line_item.internal_buyer_line_item_id) as
aws_ref_fee_credit_disbursed,

-- aggregate multiple disbursement
max(disbursement_date) over (partition by line_item.internal_buyer_line_item_id) as
last_disbursement_date,
    first_value(case when disbursement_id_or_invoiced =
'<invoiced>' then null else disbursement_id_or_invoiced end)
over(partition by line_item.internal_buyer_line_item_id order by
coalesce(disbursement_date,cast('1900-01-01' as timestamp)) desc rows between
unbounded preceding and unbounded following) as last_disbursement_id,
    first_value(bank_trace_id) over (partition by line_item.internal_buyer_line_item_id
order by coalesce(disbursement_date,cast('1900-01-01' as timestamp)) desc rows between
unbounded preceding and unbounded following) as last_disburse_bank_trace_id,
    disb_list.disbursement_date_list,
    disb_list.disburse_bank_trace_id_list
from
    line_items_aggregated as line_item
    left join disbursement_list disb_list
        on line_item.internal_buyer_line_item_id = disb_list.internal_buyer_line_item_id
),
cppo_offer_id as (
    select

```

```

-- Channel partner offers do not exist in offertargetfeed_v1 table (as per legal
requirement), causing cppo offer be defined as 'Public' in previous step, we will
convert them back to 'Private' in next step
  offer_id
from
  offers_with_uni_temporal_data
where
  -- seller_account_id is null means the ISV owns the offer
  seller_account_id is not null
  and seller_account_id '<>' (select seller_account_id from seller_account)
group by
  offer_id
),
line_items_with_window_functions_enrich_offer_product_address as (
  select
    internal_buyer_line_item_id,
    disbursement_id,
    disbursement_id_or_invoiced,
    line.product_id,
    legacy_product.legacy_id as legacy_product_id,
    products.title as product_title,
    line.broker_id,
    line.currency,
    line.end_user_account_id,
    acc_enduser.encrypted_account_id as end_user_encrypted_account_id,
    acc_enduser.aws_account_id as end_user_aws_account_id,
    acc_payer.aws_account_id as payer_aws_account_id,
    acc_payer.encrypted_account_id payer_encrypted_account_id,
    line.agreement_id,
    agreement.agreement_revision,
    line.proposer_account_id,
    case when offer.offer_id like 'aiqoffer-%' then null else agreement.start_date end
as Agreement_Start_Date,
    case when offer.offer_id like 'aiqoffer-%' then null else agreement.end_date end as
Agreement_End_Date,
    case when offer.offer_id like 'aiqoffer-%' then null else agreement.acceptance_date
end as Agreement_Acceptance_Date,
    case when offer.offer_id like 'aiqoffer-%' then null else agreement.valid_from end
as agreement_updated_date,
    case when offer.offer_id like 'aiqoffer-%' then null else
line.usage_period_start_date end as Usage_Period_Start_Date,
    case when offer.offer_id like 'aiqoffer-%' then null else
line.usage_period_end_date end as Usage_Period_End_Date,

```

```
line.acceptor_account_id,  
acc_subscriber.aws_account_id as subscriber_aws_account_id,  
acc_subscriber.encrypted_account_id as subscriber_encrypted_account_id,  
offer.offer_id,  
case  
  when offer.offer_id in (  
    select distinct offer_id  
    from cppo_offer_id)  
  then 'Private'  
  else offer.offer_target  
end as offer_target,  
offer.name offer_name,  
offer.opportunity_name offer_opportunity_name,  
offer.opportunity_description offer_opportunity_description,  
offer.opportunity_id,  
payment_due_date,  
line.bank_trace_id,  
disbursement_date,  
billing_address_id,  
buyer_invoice_id,  
seller_invoice_id,  
buyer_invoice_date,  
seller_invoice_date,  
gross_revenue_this_disbursement_id_or_invoiced,  
gross_refund_this_disbursement_id_or_invoiced,  
cogs_this_disbursement_id_or_invoiced,  
cogs_refund_this_disbursement_id_or_invoiced,  
aws_rev_share_this_disbursement_id_or_invoiced,  
aws_refund_share_this_disbursement_id_or_invoiced,  
aws_tax_share_this_disbursement_id_or_invoiced,  
aws_tax_share_listing_fee_this_disbursement_id_or_invoiced,  
aws_tax_share_refund_this_disbursement_id_or_invoiced,  
aws_tax_share_refund_listing_fee_this_disbursement_id_or_invoiced,  
seller_tax_share_this_disbursement_id_or_invoiced,  
seller_tax_share_refund_this_disbursement_id_or_invoiced,  
balance_adjustment_this_disbursement_id_or_invoiced,  
seller_rev_credit_this_disbursement_id_or_invoiced,  
aws_ref_fee_credit_this_disbursement_id_or_invoiced,  
gross_revenue_invoiced,  
gross_refund_invoiced,  
cogs_invoiced,  
cogs_refund_invoiced,  
aws_rev_share_invoiced,
```

```
aws_refund_share_invoiced,
aws_tax_share_invoiced,
aws_tax_share_listing_fee_invoiced,
aws_tax_share_refund_invoiced,
aws_tax_share_refund_listing_fee_invoiced,
seller_tax_share_invoiced,
seller_tax_share_refund_invoiced,
balance_adjustment_invoiced,
seller_rev_credit_invoiced,
aws_ref_fee_credit_invoiced,
gross_revenue_disbursed,
gross_refund_disbursed,
cogs_disbursed,
cogs_refund_disbursed,
aws_rev_share_disbursed,
aws_refund_share_disbursed,
aws_tax_share_disbursed,
aws_tax_share_listing_fee_disbursed,
aws_tax_share_refund_disbursed,
aws_tax_share_refund_listing_fee_disbursed,
seller_tax_share_disbursed,
seller_tax_share_refund_disbursed,
balance_adjustment_disbursed,
seller_rev_credit_disbursed,
aws_ref_fee_credit_disbursed,
last_disbursement_date,
last_disbursement_id,
last_disburse_bank_trace_id,
disbursement_date_list,
disburse_bank_trace_id_list,
products.product_code,
acc_products.aws_account_id as manufacturer_aws_account_id,
products.manufacturer_account_id,
--add subscriber and payer addressID, payer address preference order: tax
address>billing address>mailing address, subscriber address preference order: tax
address> mailing address
coalesce (
  --empty value in Athena shows as '', change all '' value to null in order to
follow the preference order logic above
  case when acc_subscriber.tax_address_id = '' then null else
acc_subscriber.tax_address_id end,
  case when acc_subscriber.mailing_address_id = '' then null else
acc_subscriber.mailing_address_id end) as subscriber_address_id,
coalesce (
```

```

        case when acc_payer.tax_address_id = '' then null else acc_payer.tax_address_id
end,
        case when line.billing_address_id = '' then null else line.billing_address_id
end,
        case when acc_payer.mailing_address_id = '' then null else
acc_payer.mailing_address_id end) as payer_address_id,
        coalesce (
            case when acc_enduser.tax_address_id = '' then null else
acc_enduser.tax_address_id end,
            case when line.billing_address_id = '' then null else line.billing_address_id
end,
            case when acc_enduser.mailing_address_id = '' then null else
acc_enduser.mailing_address_id end) as end_user_address_id
from
    line_items_with_window_functions as line
left join agreements_with_history as agreement on
    line.agreement_id = agreement.agreement_id and line.buyer_invoice_date >=
agreement.valid_from_adjusted and line.buyer_invoice_date < agreement.valid_to
left join offers_with_history_with_target_type as offer on
    line.offer_id = offer.offer_id and line.buyer_invoice_date >= offer.valid_from
and line.buyer_invoice_date < offer.valid_to
left join products_with_history as products on
    line.product_id = products.product_id and line.buyer_invoice_date >=
products.valid_from_adjusted and line.buyer_invoice_date < products.valid_to
left join legacy_products as legacy_product on
    line.product_id = legacy_product.new_id
left join accounts_with_history_with_company_name as acc_payer on
    line.payer_account_id = acc_payer.account_id and line.buyer_invoice_date >=
acc_payer.valid_from and line.buyer_invoice_date < acc_payer.valid_to
left join accounts_with_history_with_company_name as acc_enduser on
    line.end_user_account_id = acc_enduser.account_id and line.buyer_invoice_date
>= acc_enduser.valid_from and line.buyer_invoice_date < acc_enduser.valid_to
left join accounts_with_history_with_company_name as acc_subscriber on
    line.acceptor_account_id = acc_subscriber.account_id and
line.buyer_invoice_date >= acc_subscriber.valid_from and line.buyer_invoice_date <
acc_subscriber.valid_to
left join accounts_with_history_with_company_name as acc_products on
    products.manufacturer_account_id = acc_products.account_id and
line.buyer_invoice_date >= acc_products.valid_from and line.buyer_invoice_date <
acc_products.valid_to
),
line_items_with_window_functions_enrich_offer_product_address_name as (

```

```
select
  line.internal_buyer_line_item_id,
  disbursement_id,
  disbursement_id_or_invoiced,
  product_id,
  legacy_product_id,
  product_title,
  broker_id,
  currency,
  end_user_address_id,
  end_user_account_id,
  end_user_encrypted_account_id,
  end_user_aws_account_id,
  add_enduser.company_name end_user_company_name,
  add_enduser.email_domain end_user_email_domain,
  add_enduser.city end_user_city,
  add_enduser.state_or_region end_user_state,
  add_enduser.country_code end_user_country,
  add_enduser.postal_code end_user_postal_code,
  payer_aws_account_id,
  payer_encrypted_account_id,
  payer_address_id,
  add_payer.company_name payer_company_name,
  add_payer.email_domain payer_email_domain,
  add_payer.city payer_city,
  add_payer.state_or_region payer_state,
  add_payer.country_code payer_country,
  add_payer.postal_code payer_postal_code,
  agreement_id,
  agreement_revision,
  agreement_start_date,
  agreement_end_date,
  agreement_acceptance_date,
  agreement_updated_date,
  case when proposer_account_id = (select seller_account_id from seller_account) then
  null else acc_proposer.aws_account_id end as reseller_aws_account_id,
  case when proposer_account_id = (select seller_account_id from seller_account) then
  null else acc_proposer.mailing_company_name end as reseller_company_name,
  usage_period_start_date,
  usage_period_end_date,
  proposer_account_id,
  acc_proposer.aws_account_id as proposer_aws_account_id,
  acceptor_account_id,
  subscriber_aws_account_id,
```

```
subscriber_encrypted_account_id,  
subscriber_address_id,  
add_subscriber.company_name subscriber_company_name,  
add_subscriber.email_domain subscriber_email_domain,  
add_subscriber.city subscriber_city,  
add_subscriber.state_or_region subscriber_state,  
add_subscriber.country_code subscriber_country,  
add_subscriber.postal_code subscriber_postal_code,  
offer_id,  
offer_target,  
offer_name,  
offer_opportunity_name,  
offer_opportunity_description,  
opportunity_id,  
payment_due_date,  
bank_trace_id,  
disbursement_date,  
billing_address_id,  
max(buyer_invoice_id)as buyer_invoice_id,  
max(seller_invoice_id)as seller_invoice_id,  
max(buyer_invoice_date)as buyer_invoice_date,  
max(seller_invoice_date)as seller_invoice_date,  
gross_revenue_this_disbursement_id_or_invoiced,  
gross_refund_this_disbursement_id_or_invoiced,  
cogs_this_disbursement_id_or_invoiced,  
cogs_refund_this_disbursement_id_or_invoiced,  
aws_rev_share_this_disbursement_id_or_invoiced,  
aws_refund_share_this_disbursement_id_or_invoiced,  
aws_tax_share_this_disbursement_id_or_invoiced,  
aws_tax_share_listing_fee_this_disbursement_id_or_invoiced,  
aws_tax_share_refund_this_disbursement_id_or_invoiced,  
aws_tax_share_refund_listing_fee_this_disbursement_id_or_invoiced,  
seller_tax_share_this_disbursement_id_or_invoiced,  
seller_tax_share_refund_this_disbursement_id_or_invoiced,  
balance_adjustment_this_disbursement_id_or_invoiced,  
seller_rev_credit_this_disbursement_id_or_invoiced,  
aws_ref_fee_credit_this_disbursement_id_or_invoiced,  
(gross_revenue_this_disbursement_id_or_invoiced +  
gross_refund_this_disbursement_id_or_invoiced +  
aws_rev_share_this_disbursement_id_or_invoiced +  
aws_refund_share_this_disbursement_id_or_invoiced +  
seller_tax_share_this_disbursement_id_or_invoiced +  
seller_tax_share_refund_this_disbursement_id_or_invoiced
```

```

    + cogs_this_disbursement_id_or_invoiced +
cogs_refund_this_disbursement_id_or_invoiced +
aws_tax_share_listing_fee_this_disbursement_id_or_invoiced +
aws_tax_share_refund_listing_fee_this_disbursement_id_or_invoiced) as
seller_net_revenue_this_disbursement_id_or_invoiced,
    gross_revenue_invoiced,
    gross_refund_invoiced,
    cogs_invoiced,
    cogs_refund_invoiced,
    aws_rev_share_invoiced,
    aws_refund_share_invoiced,
    aws_tax_share_invoiced,
    aws_tax_share_listing_fee_invoiced,
    aws_tax_share_refund_invoiced,
    aws_tax_share_refund_listing_fee_invoiced,
    seller_tax_share_invoiced,
    seller_tax_share_refund_invoiced,
    balance_adjustment_invoiced,
    seller_rev_credit_invoiced,
    aws_ref_fee_credit_invoiced,
    gross_revenue_disbursed,
    gross_refund_disbursed,
    cogs_disbursed,
    cogs_refund_disbursed,
    aws_rev_share_disbursed,
    aws_refund_share_disbursed,
    aws_tax_share_disbursed,
    aws_tax_share_listing_fee_disbursed,
    aws_tax_share_refund_disbursed,
    aws_tax_share_refund_listing_fee_disbursed,
    seller_tax_share_disbursed,
    seller_tax_share_refund_disbursed,
    balance_adjustment_disbursed,
    seller_rev_credit_disbursed,
    aws_ref_fee_credit_disbursed,
    (gross_revenue_invoiced + gross_revenue_disbursed) as uncollected_gross_revenue,
    -- net revenue = gross revenue - listing fee - tax - cogs
    (gross_revenue_invoiced + gross_refund_invoiced + aws_rev_share_invoiced
+ aws_refund_share_invoiced + seller_tax_share_invoiced +
seller_tax_share_refund_invoiced + cogs_invoiced + cogs_refund_invoiced +
aws_tax_share_listing_fee_invoiced + aws_tax_share_refund_listing_fee_invoiced) as
seller_net_revenue,
    (gross_revenue_invoiced + gross_refund_invoiced + aws_rev_share_invoiced
+ aws_refund_share_invoiced + seller_tax_share_invoiced +

```

```

seller_tax_share_refund_invoiced + cogs_invoiced + cogs_refund_invoiced +
aws_tax_share_listing_fee_invoiced + aws_tax_share_refund_listing_fee_invoiced
  + gross_revenue_disbursed + gross_refund_disbursed + aws_rev_share_disbursed
+ aws_refund_share_disbursed + seller_tax_share_disbursed +
seller_tax_share_refund_disbursed + cogs_disbursed + cogs_refund_disbursed +
aws_tax_share_listing_fee_disbursed + aws_tax_share_refund_listing_fee_disbursed) as
uncollected_seller_net_revenue,
  last_disbursement_date,
  last_disbursement_id,
  last_disburse_bank_trace_id,
  disbursement_date_list,
  disburse_bank_trace_id_list,
  product_code,
  manufacturer_aws_account_id,
  manufacturer_account_id,
  acc_manu.mailing_company_name as manufacturer_company_name,
  cast(null as varchar) as AR_Period,
  case
    when (
      (gross_revenue_invoiced '<>0 and gross_revenue_invoiced = -1 *
gross_revenue_disbursed)
      or (gross_refund_invoiced '<> 0 and gross_refund_invoiced = -1 *
gross_refund_disbursed)
      or (balance_adjustment_invoiced '<> 0 and balance_adjustment_invoiced = -1 *
balance_adjustment_disbursed)
      or (seller_tax_share_refund_invoiced '<> 0 and seller_tax_share_refund_invoiced
= -1 * seller_tax_share_refund_disbursed)
      or (gross_revenue_invoiced = 0 and gross_refund_invoiced = 0 and
balance_adjustment_invoiced = 0 and seller_tax_share_refund_invoiced = 0 and
last_disbursement_id is not null)) then 'Yes'
      when gross_revenue_disbursed = 0 and gross_refund_disbursed = 0 and
balance_adjustment_disbursed = 0 and seller_tax_share_disbursed = 0 and
seller_tax_share_refund_disbursed = 0 then 'No'
      else 'Partial'
    end as Disbursement_Flag
  from line_items_with_window_functions_enrich_offer_product_address as line
  left join accounts_with_history_with_company_name as acc_manu on
    line.manufacturer_account_id = acc_manu.account_id and line.buyer_invoice_date >=
acc_manu.valid_from_adjusted and line.buyer_invoice_date <= acc_manu.valid_to
  left join accounts_with_history_with_company_name as acc_proposer on
    line.proposer_account_id = acc_proposer.account_id and line.buyer_invoice_date >=
acc_proposer.valid_from and line.buyer_invoice_date < acc_proposer.valid_to
  left join address_with_latest_revision as add_payer on
    line.payer_address_id = add_payer.address_id

```

```
left join address_with_latest_revision as add_subscriber on
  line.subscriber_address_id = add_subscriber.address_id
left join address_with_latest_revision as add_enduser on
  line.end_user_address_id = add_enduser.address_id
group by
  line.internal_buyer_line_item_id,
  disbursement_id,
  disbursement_id_or_invoiced,
  product_id,
  legacy_product_id,
  product_title,
  broker_id,
  currency,
  end_user_address_id,
  end_user_account_id,
  end_user_encrypted_account_id,
  end_user_aws_account_id,
  add_enduser.company_name,
  add_enduser.email_domain,
  add_enduser.city,
  add_enduser.state_or_region,
  add_enduser.country_code,
  add_enduser.postal_code,
  payer_aws_account_id,
  payer_encrypted_account_id,
  payer_address_id,
  add_payer.company_name,
  add_payer.email_domain,
  add_payer.city,
  add_payer.state_or_region,
  add_payer.country_code,
  add_payer.postal_code,
  agreement_id,
  agreement_revision,
  case when proposer_account_id = (select seller_account_id from seller_account) then
null else acc_proposer.aws_account_id end,
  case when proposer_account_id = (select seller_account_id from seller_account) then
null else acc_proposer.mailing_company_name end,
  agreement_start_date,
  agreement_end_date,
  agreement_acceptance_date,
  agreement_updated_date,
  usage_period_start_date,
  usage_period_end_date,
```

```
acceptor_account_id,  
subscriber_aws_account_id,  
subscriber_encrypted_account_id,  
subscriber_address_id,  
add_subscriber.company_name,  
add_subscriber.email_domain,  
add_subscriber.city,  
add_subscriber.state_or_region,  
add_subscriber.country_code,  
add_subscriber.postal_code,  
offer_id,  
offer_target,  
offer_name,  
offer_opportunity_name,  
offer_opportunity_description,  
opportunity_id,  
payment_due_date,  
bank_trace_id,  
disbursement_date,  
billing_address_id,  
gross_revenue_this_disbursement_id_or_invoiced,  
gross_refund_this_disbursement_id_or_invoiced,  
cogs_this_disbursement_id_or_invoiced,  
cogs_refund_this_disbursement_id_or_invoiced,  
aws_rev_share_this_disbursement_id_or_invoiced,  
aws_refund_share_this_disbursement_id_or_invoiced,  
aws_tax_share_this_disbursement_id_or_invoiced,  
aws_tax_share_listing_fee_this_disbursement_id_or_invoiced,  
aws_tax_share_refund_this_disbursement_id_or_invoiced,  
aws_tax_share_refund_listing_fee_this_disbursement_id_or_invoiced,  
seller_tax_share_this_disbursement_id_or_invoiced,  
seller_tax_share_refund_this_disbursement_id_or_invoiced,  
balance_adjustment_this_disbursement_id_or_invoiced,  
seller_rev_credit_this_disbursement_id_or_invoiced,  
aws_ref_fee_credit_this_disbursement_id_or_invoiced,  
gross_revenue_invoiced,  
gross_refund_invoiced,  
cogs_invoiced,  
cogs_refund_invoiced,  
aws_rev_share_invoiced,  
aws_refund_share_invoiced,  
aws_tax_share_invoiced,  
aws_tax_share_listing_fee_invoiced,  
aws_tax_share_refund_invoiced,
```

```
aws_tax_share_refund_listing_fee_invoiced,
seller_tax_share_invoiced,
seller_tax_share_refund_invoiced,
balance_adjustment_invoiced,
seller_rev_credit_invoiced,
aws_ref_fee_credit_invoiced,
gross_revenue_disbursed,
gross_refund_disbursed,
cogs_disbursed,
cogs_refund_disbursed,
aws_rev_share_disbursed,
aws_refund_share_disbursed,
aws_tax_share_disbursed,
aws_tax_share_listing_fee_disbursed,
aws_tax_share_refund_disbursed,
aws_tax_share_refund_listing_fee_disbursed,
seller_tax_share_disbursed,
seller_tax_share_refund_disbursed,
balance_adjustment_disbursed,
seller_rev_credit_disbursed,
aws_ref_fee_credit_disbursed,
last_disbursement_date,
last_disbursement_id,
last_disburse_bank_trace_id,
disbursement_date_list,
disburse_bank_trace_id_list,
product_code,
manufacturer_aws_account_id,
manufacturer_account_id,
acc_manu.mailing_company_name,
proposer_account_id,
acc_proposer.aws_account_id
),
invoiced_not_disbursed as(
  select
    --we will filter on rownum =1 in next step,
    -- means internal_buyer_line_item_id, there's only '<invoiced>' record, no
    disbursement_id linked
    *,
    max(case when disbursement_id_or_invoiced = '<invoiced>' then 1 else 2 end)
      over (partition by internal_buyer_line_item_id) rownum
  from line_items_with_window_functions_enrich_offer_product_address_name as line_items
),
```

```

collections_and_disbursements as (
  select
    -----
    -- Invoice Info --
    -----
    buyer_invoice_date as Invoice_Date,
    Payment_Due_Date as Payment_Due_Date,
    concat(
      'Net ',
      case
        when abs(date_diff('Day', Payment_due_date, buyer_invoice_date))>180 then
'180+'
        else cast(abs(date_diff('Day', Payment_due_date, buyer_invoice_date)) as
varchar)
      end,
      ' days'
    ) as payment_terms,
    buyer_invoice_id as Invoice_ID,
    coalesce(
      --empty value in Athena shows as '', change all '' value to null
      case when seller_invoice_id = '' then null else seller_invoice_id end,
      'Not applicable') as Listing_Fee_Invoice_ID,

    -----
    --End user Information --
    -----
    coalesce(
      --empty value in Athena shows as '', change all '' value to null
      case when End_User_Company_Name = '' then null else End_User_Company_Name end,
      'Not available') as End_User_Company_Name,
    End_User_AWS_Account_ID,
    End_User_Encrypted_Account_ID,
    End_User_Email_Domain,
    End_User_City,
    End_User_State as End_User_State_or_Region,
    End_User_Country,
    End_User_Postal_Code,
    End_User_Address_ID,

    -----
    --Subscriber Information --
    -----
    case
      when Agreement_Id is null or Agreement_ID = '' then 'Not available'

```

```
    when Subscriber_Company_Name is null or Subscriber_Company_Name = '' then 'Not
provided'
    else Subscriber_Company_Name
end as Subscriber_Company_Name,
case
    when Agreement_Id is null or Agreement_ID = '' then 'Not available'
    else Subscriber_AWS_Account_ID
end as Subscriber_AWS_Account_ID,
case
    when Agreement_Id is null or Agreement_ID = '' then 'Not available'
    else Subscriber_Encrypted_Account_ID
end as Subscriber_Encrypted_Account_ID,
case
    when Agreement_Id is null or Agreement_ID = '' then 'Not available'
    when Subscriber_Email_Domain is null or Subscriber_Email_Domain = '' then 'Not
provided'
    else Subscriber_Email_Domain
end as Subscriber_Email_Domain,
case
    when Agreement_id is null or Agreement_ID = '' then 'Not available'
    when Subscriber_City is null or Subscriber_City = '' then 'Not provided'
    else Subscriber_City
end as Subscriber_City,
case
    when Agreement_Id is null or Agreement_ID = '' then 'Not available'
    when Subscriber_State is null or Subscriber_State = '' then 'Not provided'
    else Subscriber_State
end as Subscriber_State_or_Region,
case
    when Agreement_Id is null or Agreement_ID = '' then 'Not available'
    when Subscriber_Country is null or Subscriber_Country = '' then 'Not provided'
    else Subscriber_Country
end as Subscriber_Country,
case
    when Agreement_Id is null or Agreement_ID = '' then 'Not available'
    when Subscriber_Postal_Code is null or Subscriber_Postal_Code = '' then 'Not
provided'
    else Subscriber_Postal_Code
end as Subscriber_Postal_Code,
case
    when Agreement_ID is null or Agreement_ID = '' then 'Not available'
    when Subscriber_Address_ID is null or Subscriber_Address_ID = '' then 'Not
provided'
    else Subscriber_Address_ID
```

```
end as Subscriber_Address_ID,

-----
-- Procurement Info --
-----
-- product title at time of invoice. It is possible that the title changes over
time and therefore there may be multiple product titles mapped to a single product id.
coalesce(
  --empty value in Athena shows as '', change all '' value to null
  case when Product_Title = '' then null else Product_Title end,
  'Not provided') as Product_Title,
-- offer name at time of invoice. It is possible that the name changes over time
therefore there may be multiple offer names mapped to a single offer id.
case
  when Agreement_Id is null or Agreement_ID = '' then 'Not available'
  when (Offer_Name is null or Offer_Name = '') and Offer_Target = 'Public' then
'Not applicable'
  else Offer_Name
end as Offer_Name,
case
  when Agreement_Id is null or Agreement_ID = ''
  then 'Not available'
  else Offer_ID
end as Offer_ID,
-- offer visibility at time of invoice.,
case
  when Agreement_Id is null or Agreement_ID = '' then 'Not available'
  else Offer_Target
end as Offer_Visibility,
coalesce(
  --empty value in Athena shows as '', change all '' value to null
  case when Agreement_ID = '' then null else Agreement_ID end,
  'Not available') as Agreement_ID,
Agreement_Start_Date,
Agreement_Acceptance_Date,
Agreement_End_Date,

Usage_Period_Start_Date,
Usage_Period_End_Date,

-----
-- Disbursement Info --
-----
case
```

```

    when Disbursement_Flag = 'Yes' then 'Disbursed'
    when Disbursement_Flag = 'No' then 'Not Disbursed'
    else 'Other'
end as Disbursement_Status,
last_disbursement_date as disbursement_date,
case
    when Disbursement_Flag = 'Yes' then date_diff('DAY',
date_trunc('DAY',payment_due_date), date_trunc('DAY',last_disbursement_date))
    else null
end as Disbursement_Time,
case
    when Disbursement_Flag = 'No' then 'Not applicable'
    when bank_trace_id is null or bank_trace_id = '' then 'Not available'
    else bank_trace_id
end as disburse_bank_trace_id,

-----
-- Revenues --
-----

-- We are rounding the sums using 2 decimal precision
-- Note that the rounding method might differ between SQL implementations.
-- The monthly revenue report is using RoundingMode.HALF_UP. This might create tiny
discrepancies between this SQL output
-- and the legacy report
round(-1 * gross_revenue_this_disbursement_id_or_invoiced,2) as Gross_Revenue,
round(-1 * gross_refund_this_disbursement_id_or_invoiced,2) as Gross_Refund,
round(-1 * aws_rev_share_this_disbursement_id_or_invoiced,2) as Listing_Fee,
round(-1 * aws_refund_share_this_disbursement_id_or_invoiced,2) as
Listing_Fee_Refund,
truncate(
    case
        when gross_revenue_this_disbursement_id_or_invoiced !
= 0 then abs(aws_rev_share_this_disbursement_id_or_invoiced/
gross_revenue_this_disbursement_id_or_invoiced)
        when gross_refund_this_disbursement_id_or_invoiced !=
0 then abs(aws_refund_share_this_disbursement_id_or_invoiced/
gross_refund_this_disbursement_id_or_invoiced)
        else 0
    end
    ,4) as Listing_Fee_Percentage,
round(-1 * seller_tax_share_this_disbursement_id_or_invoiced,2) as
Seller_Tax_Share,
round(-1 * seller_tax_share_refund_this_disbursement_id_or_invoiced,2) as
Seller_Tax_Share_Refund,

```

```

    round(-1 * aws_tax_share_listing_fee_this_disbursement_id_or_invoiced,2) as
AWS_Tax_Share_Listing_Fee,
    round(-1 * aws_tax_share_refund_listing_fee_this_disbursement_id_or_invoiced,2) as
AWS_Tax_Share_Refund_Listing_Fee,
    round(-1 * cogs_this_disbursement_id_or_invoiced,2) as Wholesale_cost,
    round(-1 * cogs_refund_this_disbursement_id_or_invoiced,2) as
Wholesale_cost_Refund,
    round(-1 * seller_net_revenue_this_disbursement_id_or_invoiced,2) as
Seller_Net_Revenue,
    currency as Currency,

    substring(internal_buyer_line_item_id,1,strpos(internal_buyer_line_item_id,'-')-1)
as Transaction_Reference_ID,
    broker_id as AWS_seller_of_record,

-----
-- Resale info --
-----
case
    when Opportunity_Id is null or Opportunity_Id = '' then
        case
            when Offer_Target = 'Public' then 'Not applicable'
            when (Offer_Target is null or Offer_Target = '') and (Agreement_Id is not
null and Agreement_Id != '') then 'Not applicable'
            else null
        end
    else Opportunity_Id
end as Resale_authorization_ID,
case
    when Offer_Opportunity_Name is null or Offer_Opportunity_Name = '' then
        case
            when Offer_Target = 'Public' then 'Not applicable'
            when (Offer_Target is null or Offer_Target = '') and (Agreement_Id is not
null and Agreement_Id != '') then 'Not applicable'
            else null
        end
    else Offer_Opportunity_Name
end as Resale_authorization_name,
case
    when Offer_Opportunity_Description is null or Offer_Opportunity_Description = ''
then
        case
            when Offer_Target = 'Public' then 'Not applicable'

```

```

        when (Offer_Target is null or Offer_Target = '') and (Agreement_Id is not
null and Agreement_Id != '') then 'Not applicable'
        else null
    end
    else Offer_Opportunity_Description
end as Resale_authorization_description,
case
    when (Reseller_AWS_Account_ID is not null and Reseller_AWS_Account_ID != '')
        and (Reseller_Company_Name is null or Reseller_Company_Name = '') then 'Not
available'
    when (Reseller_AWS_Account_ID is null or Reseller_AWS_Account_ID = '')
        and (opportunity_id is null or opportunity_id = '') then 'Not applicable'
    when (select seller_account_id from seller_account) '<>
manufacturer_aws_account_id
        and (Reseller_AWS_Account_ID is null or Reseller_AWS_Account_ID = '') then 'Not
applicable'
    else Reseller_Company_Name
end as Reseller_Company_Name,
case
    when (Reseller_AWS_Account_ID is null or Reseller_AWS_Account_ID = '')
        and (Opportunity_Id is null or Opportunity_Id = '') then 'Not applicable'
    when (select seller_account_id from seller_account) '<>
manufacturer_aws_account_id
        and (Reseller_AWS_Account_ID is null or Reseller_AWS_Account_ID = '') then 'Not
applicable'
    else Reseller_AWS_Account_ID
end as Reseller_AWS_Account_ID,

-----
-- Payer Information --
-----
coalesce(
    --empty value in Athena shows as '', change all '' value to null
    case when Payer_Company_Name = '' then null else Payer_Company_Name end,
    'Not available') as Payer_Company_Name,
Payer_AWS_Account_ID, -- "Customer AWS Account Number" in legacy report
Payer_Encrypted_Account_ID,
Payer_Email_Domain,
Payer_City,
Payer_State as Payer_State_or_Region,
Payer_Country,
Payer_Postal_Code,
Payer_Address_ID,

```

```

-----
-- ISV Information --
-----
manufacturer_aws_account_id as ISV_Account_ID,
coalesce(
  --empty value in Athena shows as '', change all '' value to null
  case when Manufacturer_Company_Name = '' then null else Manufacturer_Company_Name
end,
  'Not available') as ISV_Company_Name,

-----
-- Products info --
-----
Legacy_Product_ID,
coalesce(
  --empty value in Athena shows as '', change all '' value to null
  case when Product_ID = '' then null else Product_ID end,
  'Not provided') as Product_ID,
-- this is to get the legacy product id https://sim.amazon.com/issues/MP-
INSIGHTS-2561
Product_Code,

  case when Disbursement_Flag = 'Yes' then round(-1 *
seller_net_revenue_this_disbursement_id_or_invoiced,2) else 0 end as
Disbursed_Net_Revenue,
  case when Disbursement_Flag = 'No' then round(-1 *
seller_net_revenue_this_disbursement_id_or_invoiced,2) else 0 end as
Undisbursed_Net_Revenue,
  case
    when case when Disbursement_Flag = 'Yes' then date_diff('DAY',
date_trunc('DAY',payment_due_date), date_trunc('DAY',last_disbursement_date)) else
null end <= 0 then 'Not due'
    when case when Disbursement_Flag = 'Yes' then date_diff('DAY',
date_trunc('DAY',payment_due_date), date_trunc('DAY',last_disbursement_date)) else
null end <=30 then '1-30 days late'
    when case when Disbursement_Flag = 'Yes' then date_diff('DAY',
date_trunc('DAY',payment_due_date), date_trunc('DAY',last_disbursement_date)) else
null end <=60 then '31-60 days late'
    when case when Disbursement_Flag = 'Yes' then date_diff('DAY',
date_trunc('DAY',payment_due_date), date_trunc('DAY',last_disbursement_date)) else
null end <=90 then '61-90 days late'
    when case when Disbursement_Flag = 'Yes' then date_diff('DAY',
date_trunc('DAY',payment_due_date), date_trunc('DAY',last_disbursement_date)) else
null end <=120 then '91-120 days late'

```

```

        when case when Disbursement_Flag = 'Yes' then date_diff('DAY',
date_trunc('DAY',payment_due_date), date_trunc('DAY',last_disbursement_date)) else
null end >=121 then '121+ days late'
        else null
        end as Disbursement_Period
from
    line_items_with_window_functions_enrich_offer_product_address_name as line
where disbursement_id_or_invoiced != '<invoiced>'

union

select
    -----
    -- Invoice Info --
    -----
    buyer_invoice_date as Invoice_Date,
    Payment_Due_Date as Payment_Due_Date,
    concat(
        'Net ',
        case
            when abs(date_diff('Day', Payment_due_date, buyer_invoice_date)) >180 then
'180+'
            else cast(abs(date_diff('Day', Payment_due_date, buyer_invoice_date)) as
varchar)
        end,
        ' days'
    ) as payment_terms,
    buyer_invoice_id as Invoice_ID,
    coalesce(
        --empty value in Athena shows as '', change all '' value to null
        case when seller_invoice_id = '' then null else seller_invoice_id end,
        'Not applicable') as Listing_Fee_Invoice_ID,

    -----
    --End user Information --
    -----
    coalesce(
        --empty value in Athena shows as '', change all '' value to null
        case when End_User_Company_Name = '' then null else End_User_Company_Name end,
        'Not available') as End_User_Company_Name,
    End_User_AWS_Account_ID,
    End_User_Encrypted_Account_ID,
    End_User_Email_Domain,
    End_User_City,

```

```
End_User_State as End_User_State_or_Region,
End_User_Country,
End_User_Postal_Code,
End_User_Address_ID,

-----
--Subscriber Information --
-----

case
  when Agreement_Id is null or Agreement_ID = '' then 'Not available'
  when Subscriber_Company_Name is null or Subscriber_Company_Name = '' then 'Not
provided'
  else Subscriber_Company_Name
end as Subscriber_Company_Name,
case
  when Agreement_Id is null or Agreement_ID = '' then 'Not available'
  else Subscriber_AWS_Account_ID
end as Subscriber_AWS_Account_ID,
case
  when Agreement_Id is null or Agreement_ID = '' then 'Not available'
  else Subscriber_Encrypted_Account_ID
end as Subscriber_Encrypted_Account_ID,
case
  when Agreement_Id is null or Agreement_ID = '' then 'Not available'
  when Subscriber_Email_Domain is null or Subscriber_Email_Domain = '' then 'Not
provided'
  else Subscriber_Email_Domain
end as Subscriber_Email_Domain,
case
  when Agreement_id is null or Agreement_ID = '' then 'Not available'
  when Subscriber_City is null or Subscriber_City = '' then 'Not provided'
  else Subscriber_City
end as Subscriber_City,
case
  when Agreement_Id is null or Agreement_ID = '' then 'Not available'
  when Subscriber_State is null or Subscriber_State = '' then 'Not provided'
  else Subscriber_State
end as Subscriber_State_or_Region,
case
  when Agreement_Id is null or Agreement_ID = '' then 'Not available'
  when Subscriber_Country is null or Subscriber_Country = '' then 'Not provided'
  else Subscriber_Country
end as Subscriber_Country,
case
```

```

    when Agreement_Id is null or Agreement_ID = '' then 'Not available'
    when Subscriber_Postal_Code is null or Subscriber_Postal_Code = '' then 'Not
provided'
    else Subscriber_Postal_Code
end as Subscriber_Postal_Code,
case
    when Agreement_ID is null or Agreement_ID = '' then 'Not available'
    when Subscriber_Address_ID is null or Subscriber_Address_ID = '' then 'Not
provided'
    else Subscriber_Address_ID
end as Subscriber_Address_ID,

-----
-- Procurement Info --
-----

coalesce(
    --empty value in Athena shows as '', change all '' value to null
    case when Product_Title = '' then null else Product_Title end,
    'Not provided') as Product_Title,
    -- offer name at time of invoice. It is possible that the name changes over time
therefore there may be multiple offer names mapped to a single offer id.
case
    when Agreement_Id is null or Agreement_ID = '' then 'Not available'
    when (Offer_Name is null or Offer_Name = '') and Offer_Target = 'Public' then
'Not applicable'
    else Offer_Name
end as Offer_Name,
case
    when Agreement_Id is null or Agreement_ID = ''
    then 'Not available'
    else Offer_ID
end as Offer_ID,
    -- offer visibility at time of invoice.,
case
    when Agreement_Id is null or Agreement_ID = '' then 'Not available'
    else Offer_Target
end as Offer_Visibility,
coalesce(
    --empty value in Athena shows as '', change all '' value to null
    case when Agreement_ID = '' then null else Agreement_ID end,
    'Not available') as Agreement_ID,
    --case when Agreement_Id is null or Agreement_Id = '' then cast(null as timestamp)
else Agreement_Start_Date end as Agreement_Start_Date,

```

```

--case when Agreement_Id is null or Agreement_Id = '' then cast(null as timestamp)
else Agreement_End_Date end as Agreement_End_Date,
--case when Agreement_Id is null or Agreement_Id = '' then cast(null as timestamp)
else Agreement_Acceptance_Date end as Agreement_Acceptance_Date,
Agreement_Start_Date,
Agreement_Acceptance_Date,
Agreement_End_Date,

Usage_Period_Start_Date,
Usage_Period_End_Date,

-----
-- Disbursement Info --
-----
case
  when Disbursement_Flag = 'Yes' then 'Disbursed'
  when Disbursement_Flag = 'No' then 'Not Disbursed'
  else 'Other'
end as Disbursement_Status,
last_disbursement_date as disbursement_date,
case
  when Disbursement_Flag = 'Yes' then date_diff('DAY',
date_trunc('DAY',payment_due_date), date_trunc('DAY',last_disbursement_date))
  else null
end as Disbursement_Time,
case
  when Disbursement_Flag = 'No' then 'Not applicable'
  when bank_trace_id is null or bank_trace_id = '' then 'Not available'
  else bank_trace_id
end as disburse_bank_trace_id,

-----
-- Revenues --
-----
-- We are rounding the sums using 2 decimal precision
-- Note that the rounding method might differ between SQL implementations.
-- The monthly revenue report is using RoundingMode.HALF_UP. This might create tiny
discrepancies between this SQL output
-- and the legacy report
round(gross_revenue_this_disbursement_id_or_invoiced,2) as Gross_Revenue,
round(gross_refund_this_disbursement_id_or_invoiced,2) as Gross_Refund,
round(aws_rev_share_this_disbursement_id_or_invoiced,2) as Listing_Fee,
round(aws_refund_share_this_disbursement_id_or_invoiced,2) as Listing_Fee_Refund,
truncate(

```

```

    case
      when gross_revenue_this_disbursement_id_or_invoiced !
= 0 then abs(aws_rev_share_this_disbursement_id_or_invoiced/
gross_revenue_this_disbursement_id_or_invoiced)
      when gross_refund_this_disbursement_id_or_invoiced !=
0 then abs(aws_refund_share_this_disbursement_id_or_invoiced/
gross_refund_this_disbursement_id_or_invoiced)
      else 0
    end
    ,4) as Listing_Fee_Percentage,
    round(seller_tax_share_this_disbursement_id_or_invoiced,2) as Seller_Tax_Share,
    round(seller_tax_share_refund_this_disbursement_id_or_invoiced,2) as
Seller_Tax_Share_Refund,
    round(aws_tax_share_listing_fee_this_disbursement_id_or_invoiced,2) as
AWS_Tax_Share_Listing_Fee,
    round(aws_tax_share_refund_listing_fee_this_disbursement_id_or_invoiced,2) as
AWS_Tax_Share_Refund_Listing_Fee,
    round(cogs_this_disbursement_id_or_invoiced,2) as Wholesale_cost,
    round(cogs_refund_this_disbursement_id_or_invoiced,2) as Wholesale_cost_Refund,
    round(seller_net_revenue_this_disbursement_id_or_invoiced,2) as Seller_Net_Revenue,
    currency as Currency,

    substring(internal_buyer_line_item_id,1,strpos(internal_buyer_line_item_id,'-')-1)
as Transaction_Reference_ID,
    broker_id as AWS_seller_of_record,

-----
-- Resale info --
-----
case
  when Opportunity_Id is null or Opportunity_Id = '' then
    case
      when Offer_Target = 'Public' then 'Not applicable'
      when (Offer_Target is null or Offer_Target = '') and (Agreement_Id is not
null and Agreement_Id != '') then 'Not applicable'
      else null
    end
  else Opportunity_Id
end as Resale_authorization_ID,
case
  when Offer_Opportunity_Name is null or Offer_Opportunity_Name = '' then
    case
      when Offer_Target = 'Public' then 'Not applicable'

```

```

        when (Offer_Target is null or Offer_Target = '') and (Agreement_Id is not
null and Agreement_Id != '') then 'Not applicable'
        else null
    end
    else Offer_Opportunity_Name
end as Resale_authorization_name,
case
    when Offer_Opportunity_Description is null or Offer_Opportunity_Description = ''
then
    case
        when Offer_Target = 'Public' then 'Not applicable'
        when (Offer_Target is null or Offer_Target = '') and (Agreement_Id is not
null and Agreement_Id != '') then 'Not applicable'
        else null
    end
    else Offer_Opportunity_Description
end as Resale_authorization_description,
case
    when (Reseller_AWS_Account_ID is not null and Reseller_AWS_Account_ID != '')
and (Reseller_Company_Name is null or Reseller_Company_Name = '') then 'Not
available'
    when (Reseller_AWS_Account_ID is null or Reseller_AWS_Account_ID = '')
and (opportunity_id is null or opportunity_id = '') then 'Not applicable'
    when (select seller_account_id from seller_account) '<>
manufacturer_aws_account_id
and (Reseller_AWS_Account_ID is null or Reseller_AWS_Account_ID = '') then 'Not
applicable'
    else Reseller_Company_Name
end as Reseller_Company_Name,
case
    when (Reseller_AWS_Account_ID is null or Reseller_AWS_Account_ID = '')
and (Opportunity_Id is null or Opportunity_Id = '') then 'Not applicable'
    when (select seller_account_id from seller_account) '<>
manufacturer_aws_account_id
and (Reseller_AWS_Account_ID is null or Reseller_AWS_Account_ID = '') then 'Not
applicable'
    else Reseller_AWS_Account_ID
end as Reseller_AWS_Account_ID,

-----
-- Payer Information --
-----

coalesce(
    --empty value in Athena shows as '', change all '' value to null

```

```

    case when Payer_Company_Name = '' then null else Payer_Company_Name end,
    'Not available') as Payer_Company_Name,
Payer_AWS_Account_ID, -- "Customer AWS Account Number" in legacy report
Payer_Encrypted_Account_ID,
Payer_Email_Domain,
Payer_City,
Payer_State as Payer_State_or_Region,
Payer_Country,
Payer_Postal_Code,
Payer_Address_ID,

-----
-- ISV Information --
-----
manufacturer_aws_account_id as ISV_Account_ID,
coalesce(
  --empty value in Athena shows as '', change all '' value to null
  case when Manufacturer_Company_Name = '' then null else Manufacturer_Company_Name
end,
  'Not available') as ISV_Company_Name,

-----
-- Products info --
-----
-- product title at time of invoice. It is possible that the title changes over
time and therefore there may be multiple product titles mapped to a single product id.
Legacy_Product_ID,
coalesce(
  --empty value in Athena shows as '', change all '' value to null
  case when Product_ID = '' then null else Product_ID end,
  'Not provided') as Product_ID,
-- this is to get the legacy product id https://sim.amazon.com/issues/MP-
INSIGHTS-2561
Product_Code,

    case when Disbursement_Flag = 'Yes' then
round(seller_net_revenue_this_disbursement_id_or_invoiced,2) else 0 end as
Disbursed_Net_Revenue,
    case when Disbursement_Flag = 'No' then
round(seller_net_revenue_this_disbursement_id_or_invoiced,2) else 0 end as
Undisbursed_Net_Revenue,
    case

```

```

        when case when Disbursement_Flag = 'Yes' then date_diff('DAY',
date_trunc('DAY',payment_due_date), date_trunc('DAY',last_disbursement_date)) else
null end <= 0 then 'Not due'
        when case when Disbursement_Flag = 'Yes' then date_diff('DAY',
date_trunc('DAY',payment_due_date), date_trunc('DAY',last_disbursement_date)) else
null end <=30 then '1-30 days late'
        when case when Disbursement_Flag = 'Yes' then date_diff('DAY',
date_trunc('DAY',payment_due_date), date_trunc('DAY',last_disbursement_date)) else
null end <=60 then '31-60 days late'
        when case when Disbursement_Flag = 'Yes' then date_diff('DAY',
date_trunc('DAY',payment_due_date), date_trunc('DAY',last_disbursement_date)) else
null end <=90 then '61-90 days late'
        when case when Disbursement_Flag = 'Yes' then date_diff('DAY',
date_trunc('DAY',payment_due_date), date_trunc('DAY',last_disbursement_date)) else
null end <=120 then '91-120 days late'
        when case when Disbursement_Flag = 'Yes' then date_diff('DAY',
date_trunc('DAY',payment_due_date), date_trunc('DAY',last_disbursement_date)) else
null end >=121 then '121+ days late'
        else null
    end as Disbursement_Period
from
    invoiced_not_disbursed
where rownum = 1

)

select *
from collections_and_disbursements
where payment_due_date >= date_add('DAY', -90, current_date)
--where payment_due_date between cast('2023-01-01' as timestamp) and cast('2024-12-31'
as timestamp)

--where disbursement_date >= date_add('DAY', -90, current_date)
--where disbursement_date between cast('2023-01-01' as timestamp) and cast('2024-12-31'
as timestamp)

```

Faktur pajak

Untuk menemukan faktur pajak Anda, Anda dapat menjalankan serangkaian kueri seperti contoh berikut. Kueri dibangun di atas satu sama lain untuk membuat laporan Perpajakan. Anda dapat

menggunakan contoh seperti yang ditunjukkan, atau menyesuaikannya untuk data dan kasus penggunaan Anda.

Komentar dalam kueri menjelaskan apa yang dilakukan kueri, dan bagaimana memodifikasinya.

```
-- Taxation report

-- General note: When executing this query we are assuming that the data ingested in
  the database is using
-- two time axes (the valid_from column and the update_date column).
-- See documentation for more details: https://docs.aws.amazon.com/marketplace/latest/userguide/data-feed.html#data-feed-details

-- An account_id has several valid_from dates (each representing a separate revision of
  the data)
-- but because of bi-temporality, an account_id + valid_from tuple can appear multiple
  times with a different update_date.
-- We are only interested in the most recent tuple (ie, uni-temporal model)
with accounts_with_uni_temporal_data as (
  select
    account_id,
    aws_account_id,
    encrypted_account_id,
    mailing_address_id,
    tax_address_id,
    tax_legal_name,
    from_iso8601_timestamp(valid_from) as valid_from,
    tax_registration_number
  from
    (
      select
        account_id,
        aws_account_id,
        encrypted_account_id,
        mailing_address_id,
        tax_address_id,
        tax_legal_name,
        valid_from,
        delete_date,
        tax_registration_number,
        row_number() over (partition by account_id, valid_from order by
  from_iso8601_timestamp(update_date) desc) as row_num
      from
        accountfeed_v1
```

```

)
where
  -- keep latest ...
  row_num = 1
  -- ... and remove the soft-deleted one.
  and (delete_date is null or delete_date = '')
),

accounts_with_history as (
  with accounts_with_history_with_extended_valid_from as (
    select
      account_id,
      -- sometimes, this columns gets imported as a "bigint" and loses heading 0s ->
      casting to a char and re-adding heading 0s (if need be)
      substring('000000000000' || cast(aws_account_id as varchar), -12) as aws_account_id,
      encrypted_account_id,
      mailing_address_id,
      tax_address_id,
      tax_legal_name tax_legal_name,
      -- The start time of account valid_from is extended to '1970-01-01 00:00:00',
      because:
      -- ... in tax report transformations, some tax line items with invoice_date
      cannot
      -- ... fall into the default valid time range of the associated account
      CASE
        WHEN LAG(valid_from) OVER (PARTITION BY account_id ORDER BY valid_from ASC) IS
        NULL
          THEN CAST('1970-01-01 00:00:00' as timestamp)
        ELSE valid_from
      END AS valid_from
    from
      (select * from accounts_with_uni_temporal_data ) as account
  )
  select
    account_id,
    aws_account_id,
    encrypted_account_id,
    mailing_address_id,
    tax_address_id,
    tax_legal_name,
    valid_from,
    coalesce(
      lead(valid_from) over (partition by account_id order by valid_from asc),
      cast('2999-01-01 00:00:00' as timestamp)

```

```
    ) as valid_to
  from
    accounts_with_history_with_extended_valid_from
),

-- A product_id has several valid_from dates (each representing a product revision),
-- but because of bi-temporality, each product_id + valid_from tuple can appear
-- multiple times with a different update_date.
-- We are only interested in the most recent tuple (ie, uni-temporal model)
products_with_uni_temporal_data as (
  select
    from_iso8601_timestamp(valid_from) as valid_from,
    from_iso8601_timestamp(update_date) as update_date,
    from_iso8601_timestamp(delete_date) as delete_date,
    product_id,
    manufacturer_account_id,
    product_code,
    title
  from
    (
      select
        valid_from,
        update_date,
        delete_date,
        product_id,
        manufacturer_account_id,
        product_code,
        title,
        row_number() over (partition by product_id, valid_from order by
from_iso8601_timestamp(update_date) desc) as row_num
      from
        productfeed_v1
    )
  where
    -- keep latest ...
    row_num = 1
    -- ... and remove the soft-deleted one.
    and (delete_date is null or delete_date = '')
),

products_with_history as (
  select
    product_id,
    title,
```

```

    valid_from,
    case
      when lag(valid_from) over (partition by product_id order by valid_from asc) is
null and valid_from < cast('2021-04-01' as timestamp)
        then date_add('Day', -3857, valid_from)
      -- 3827 is the longest delay between acceptance_date of an agreement and the
product
      -- we are keeping 3857 as a consistency between the offers and products
      when lag(valid_from) over (partition by product_id order by valid_from asc) is
null and valid_from >= cast('2021-04-01' as timestamp)
        then date_add('Day', -2190, valid_from)
      --after 2021 for the two offers we need to adjust for 2 more years
      else valid_from end as valid_from_adjusted,
    coalesce(
      lead(valid_from) over (partition by product_id order by valid_from asc),
      cast('2999-01-01 00:00:00' as timestamp)
    ) as valid_to,
    product_code,
    manufacturer_account_id
  from
    products_with_uni_temporal_data
),

-- A tax_item_id has several valid_from dates (each representing a product revision),
-- but because of bi-temporality, each tax_item_id + valid_from tuple can appear
multiple times with a different update_date.
-- We are only interested in the most recent tuple (ie, uni-temporal model)
tax_items_with_uni_temporal_data as (
  select
    from_iso8601_timestamp(valid_from) as valid_from,
    from_iso8601_timestamp(update_date) as update_date,
    delete_date,
    cast(tax_item_id as varchar) as tax_item_id,
    cast(invoice_id as varchar) as invoice_id,
    cast(line_item_id as varchar) as line_item_id,
    cast(customer_bill_id as varchar) as customer_bill_id,
    tax_liable_party,
    transaction_type_code,
    product_id,
    product_tax_code,
    from_iso8601_timestamp(invoice_date) as invoice_date,
    taxed_customer_account_id,
    taxed_customer_country,
    taxed_customer_state_or_region,

```

```
taxed_customer_city,  
taxed_customer_postal_code,  
tax_location_code_taxed_jurisdiction,  
tax_type_code,  
jurisdiction_level,  
taxed_jurisdiction,  
display_price_taxability_type,  
tax_jurisdiction_rate,  
tax_amount,  
tax_currency,  
tax_calculation_reason_code,  
date_used_for_tax_calculation,  
customer_exemption_certificate_id,  
customer_exemption_certificate_id_domain,  
customer_exemption_certificate_level,  
customer_exemption_code,  
customer_exemption_domain,  
transaction_reference_id  
from  
(  
  select  
    valid_from,  
    update_date,  
    delete_date,  
    tax_item_id,  
    invoice_id,  
    line_item_id,  
    customer_bill_id,  
    tax_liable_party,  
    transaction_type_code,  
    product_id,  
    product_tax_code,  
    invoice_date,  
    taxed_customer_account_id,  
    taxed_customer_country,  
    taxed_customer_state_or_region,  
    taxed_customer_city,  
    taxed_customer_postal_code,  
    tax_location_code_taxed_jurisdiction,  
    tax_type_code,  
    jurisdiction_level,  
    taxed_jurisdiction,  
    display_price_taxability_type,  
    tax_jurisdiction_rate,
```

```

        tax_amount,
        tax_currency,
        tax_calculation_reason_code,
        date_used_for_tax_calculation,
        customer_exemption_certificate_id,
        customer_exemption_certificate_id_domain,
        customer_exemption_certificate_level,
        customer_exemption_code,
        customer_exemption_domain,
        transaction_reference_id,
        row_number() over (partition by tax_item_id, valid_from order by
from_iso8601_timestamp(update_date) desc) as row_num
    from
        taxitemfeed_v1
)
where
    -- keep latest ...
    row_num = 1
    -- ... and remove the soft-deleted one.
    and (delete_date is null or delete_date = '')
),

taxation as (
    select
        tax_items.invoice_id,
        tax_items.line_item_id,
        tax_items.customer_bill_id,
        tax_items.tax_liable_party,
        tax_items.transaction_type_code,
        tax_items.product_id,
        product_tax_item.title as product_title,
        tax_items.product_tax_code,
        tax_items.invoice_date,
        accounts_with_history.aws_account_id as taxed_customer_account_id,
        tax_items.taxed_customer_country,
        tax_items.taxed_customer_state_or_region,
        tax_items.taxed_customer_city,
        tax_items.taxed_customer_postal_code,
        tax_items.tax_type_code as tax_type,
        tax_items.jurisdiction_level,
        tax_items.taxed_jurisdiction,
        tax_items.display_price_taxability_type,
        tax_items.tax_jurisdiction_rate,
        tax_items.tax_amount,

```

```
tax_items.tax_currency,  
tax_items.tax_calculation_reason_code,  
tax_items.date_used_for_tax_calculation,  
coalesce(  
  --empty value in Athena shows as '', change all '' value to null  
  case when tax_items.customer_exemption_certificate_id = '' then null else  
tax_items.customer_exemption_certificate_id end,  
  'Not exempt') customer_exemption_certificate_id,  
coalesce(--empty value in Athena shows as '', change all '' value to null  
  case when tax_items.customer_exemption_certificate_id_domain = '' then null else  
tax_items.customer_exemption_certificate_id_domain end,  
  'Not exempt') customer_exemption_certificate_id_domain,  
coalesce(--empty value in Athena shows as '', change all '' value to null  
  case when tax_items.customer_exemption_certificate_level = '' then null else  
tax_items.customer_exemption_certificate_level end,  
  'Not exempt') customer_exemption_certificate_level,  
coalesce(--empty value in Athena shows as '', change all '' value to null  
  case when tax_items.customer_exemption_code = '' then null else  
tax_items.customer_exemption_code end,  
  'Not exempt') customer_exemption_code,  
tax_items.transaction_reference_id  
from  
  tax_items_with_uni_temporal_data as tax_items  
  left join products_with_history as product_tax_item on  
    tax_items.product_id = product_tax_item.product_id and tax_items.invoice_date  
>= product_tax_item.valid_from_adjusted and tax_items.invoice_date <  
product_tax_item.valid_to  
  left join accounts_with_history as accounts_with_history on  
    tax_items.taxed_customer_account_id = accounts_with_history.account_id and  
tax_items.invoice_date >= accounts_with_history.valid_from and tax_items.invoice_date  
< accounts_with_history.valid_to  
)  
  
select *  
from taxation  
where invoice_date >= date_add('DAY', -90, current_date)  
--where invoice_date between cast('2023-01-01' as timestamp) and cast('2024-12-31' as  
timestamp)
```

Pencairan berdasarkan produk

Untuk menemukan jumlah yang dicairkan oleh produk, Anda dapat menjalankan serangkaian kueri seperti berikut ini. Contoh ini sebanding dengan laporan [Laporan pencairan](#) penjual.

Contoh kueri ini dibangun di atas satu sama lain untuk membuat daftar akhir detail produk dengan pencairan. Ini juga menunjukkan cara mendapatkan informasi produk pada titik waktu tertentu. Anda dapat menggunakan sampel ini seperti yang ditunjukkan, atau menyesuaikannya untuk data dan kasus penggunaan Anda.

Komentar dalam kueri menjelaskan apa yang dilakukan kueri, dan bagaimana memodifikasinya.

Note

Saat menjalankan kueri ini, kami berasumsi bahwa data dicerna menggunakan dua sumbu waktu, `valid_from` dan `update` kolom. Untuk informasi lebih lanjut tentang sumbu, lihat [Penyimpanan dan struktur umpan AWS Marketplace data](#).

```
-- Get all the products and keep the latest product_id, valid_from tuple
with products_with_uni_temporal_data as (
  select
    *
  from
    (
      select
        *,
        ROW_NUMBER() OVER (PARTITION BY product_id, valid_from
          ORDER BY from_iso8601_timestamp(update_date) desc)
          as row_num
      from
        productfeed_v1
    )
  where
    -- A product_id can appear multiple times with the same
    -- valid_from date but with a different update_date column,
    -- making it effectively bi-temporal. By only taking the most
    -- recent tuple, we are converting to a uni-temporal model.
    row_num = 1
),

-- Gets the latest revision of a product
```

```
-- A product can have multiple revisions where some of the
-- columns, like the title, can change.
-- For the purpose of the disbursement report, we want
-- to get the latest revision of a product
products_with_latest_version as (
  select
    *
  from
    (
      select
        *,
        ROW_NUMBER() OVER (PARTITION BY product_id
          ORDER BY from_iso8601_timestamp(valid_from) desc)
          as row_num_latest_version
      from
        products_with_uni_temporal_data
    )
  where
    row_num_latest_version = 1
),

-- Get all the accounts and keep the latest account_id, valid_from tuple
accounts_with_uni_temporal_data as (
  select
    *
  from
    (
      select
        *,
        ROW_NUMBER() OVER (PARTITION BY account_id, valid_from ORDER BY
from_iso8601_timestamp(update_date) desc) as row_num
      from
        accountfeed_v1
    )
  where
    -- An account_id can appear multiple times with the same
    -- valid_from date but with a different update_date column,
    -- making it effectively bi-temporal. By only taking the most
    -- recent tuple, we are converting to a uni-temporal model.
    row_num = 1
),

-- Gets the latest revision of an account
-- An account can have multiple revisions where some of the
```

```
-- columns, like the mailing_address_id, can change.
-- For the purpose of the disbursement report, we want
-- to get the latest revision of a product
accounts_with_latest_version as (
  select
    *
  from
    (
      select
        *,
        ROW_NUMBER() OVER (PARTITION BY account_id
          ORDER BY from_iso8601_timestamp(valid_from) desc)
          as row_num_latest_version
      from
        accounts_with_uni_temporal_data
    )
  where
    row_num_latest_version = 1
),

-- Get all the billing events and keep the
-- latest billing_event_id, valid_from tuple:
billing_events_with_uni_temporal_data as (
  select
    *
  from (
    select
      billing_event_id,
      from_iso8601_timestamp(valid_from) as valid_from,
      from_iso8601_timestamp(update_date) as update_date,
      from_iso8601_timestamp(invoice_date) as invoice_date,
      transaction_type,
      transaction_reference_id,
      product_id,
      disbursement_billing_event_id,
      action,
      from_account_id,
      to_account_id,
      end_user_account_id,
      CAST(amount as decimal(20, 10)) invoice_amount,
      bank_trace_id,
      ROW_NUMBER() OVER (PARTITION BY billing_event_id, valid_from
        ORDER BY from_iso8601_timestamp(update_date) desc)
        as row_num
```

```
        from
            billingeventfeed_v1
        )
    where row_num = 1
),

-- Get all the disbursements
-- The billing events data is immutable.
-- It is not required to use time windows based on the
-- valid_from column to get the most recent billing event
disbursement_events as (
    select
        billing_events_raw.billing_event_id as disbursement_id,
        billing_events_raw.invoice_date as disbursement_date,
        billing_events_raw.bank_trace_id
    from
        billing_events_with_uni_temporal_data billing_events_raw
    where
        -- Only interested in disbursements, so filter out
        -- non-disbursements by selecting transaction type
        -- to be DISBURSEMENT:
        billing_events_raw.transaction_type = 'DISBURSEMENT'
        -- Select a time period, you can adjust the dates
        -- below if need be. For billing events use the
        -- invoice date as the point in time of the
        -- disbursement being initiated:
        and billing_events_raw.invoice_date >=
            from_iso8601_timestamp('2020-10-01T00:00:00Z')
        and billing_events_raw.invoice_date <
            from_iso8601_timestamp('2020-11-01T00:00:00Z')
),

-- Get the invoices along with the line items that
-- are part of the above filtered disbursements
disbursed_line_items as (
    select
        line_items.transaction_reference_id,
        line_items.product_id,
        line_items.transaction_type,
        (case
            -- Get the payer of the invoice from any
            -- transaction type that is not AWS and
            -- not BALANCE_ADJUSTMENT.
            -- For AWS and BALANCE_ADJUSTMENT, the billing
```

```

        -- event feed will show the "AWS Marketplace"
        -- account as the receiver of the funds and the
        -- seller as the payer. Filter those out.
    when line_items.transaction_type
        not like '%AWS%' and transaction_type
        not like 'BALANCE_ADJUSTMENT'
        then line_items.from_account_id
end) as payer_account_id,
line_items.end_user_account_id,
invoice_amount,
disbursements.disbursement_date,
disbursements.disbursement_id,
disbursements.bank_trace_id
from
    billing_events_with_uni_temporal_data line_items
    -- Each disbursed line item is linked to the parent
    -- disbursement via the disbursement_billing_event_id
join disbursement_events disbursements
    on disbursements.disbursement_id
    = line_items.disbursement_billing_event_id
where
    -- we are interested only in the invoice line
    -- items that are DISBURSED
    line_items.action = 'DISBURSED'
),

-- An invoice can contain multiple line items
-- Create a pivot table to calculate the different
-- amounts that are part of an invoice.
-- The new row is aggregated at
-- transaction_reference_id - end_user_account_id level
invoice_amounts_aggregated as (
    select
        transaction_reference_id,
        product_id,
        -- a given disbursement id should have the
        -- same disbursement_date
        max(disbursement_date) as disbursement_date,
        -- Build a pivot table in order to provide all the
        -- data related to a transaction in a single row.
        -- Note that the amounts are negated. This is because
        -- when an invoice is generated, we give you the
        -- positive amounts and the disbursement event
        -- negates the amounts

```

```
sum(case when transaction_type = 'SELLER_REV_SHARE'
      then -invoice_amount else 0 end) as seller_rev_share,
sum(case when transaction_type = 'AWS_REV_SHARE'
      then -invoice_amount else 0 end) as aws_rev_share,
sum(case when transaction_type = 'SELLER_REV_SHARE_REFUND'
      then -invoice_amount else 0 end) as seller_rev_refund,
sum(case when transaction_type = 'AWS_REV_SHARE_REFUND'
      then -invoice_amount else 0 end) as aws_rev_refund,
sum(case when transaction_type = 'SELLER_REV_SHARE_CREDIT'
      then -invoice_amount else 0 end) as seller_rev_credit,
sum(case when transaction_type = 'AWS_REV_SHARE_CREDIT'
      then -invoice_amount else 0 end) as aws_rev_credit,
sum(case when transaction_type = 'SELLER_TAX_SHARE'
      then -invoice_amount else 0 end) as seller_tax_share,
sum(case when transaction_type = 'SELLER_TAX_SHARE_REFUND'
      then -invoice_amount else 0 end) as seller_tax_refund,
-- This is the account that pays the invoice:
max(payer_account_id) as payer_account_id,
-- This is the account that subscribed to the product:
end_user_account_id as customer_account_id,
bank_trace_id
from
  disbursed_line_items
group by
  transaction_reference_id,
  product_id,
  disbursement_id,
  -- There might be a different end-user for the same
  -- transaction reference id. Distributed licenses
  -- is an example
  end_user_account_id,
  bank_trace_id
),

disbursed_amount_by_product as (
  select
    products.title as ProductTitle,
    products.product_code as ProductCode,
    -- We are rounding the sums using 2 decimal precision
    -- Note that the rounding method might differ
    -- between SQL implementations.
    -- The disbursement seller report is using
    -- RoundingMode.HALF_UP. This might create
    -- discrepancies between this SQL output
```

```
-- and the disbursement seller report
round(invoice_amounts.seller_rev_share, 2) as SellerRev,
round(invoice_amounts.aws_rev_share, 2) as AWSRefFee,
round(invoice_amounts.seller_rev_refund, 2) as SellerRevRefund,
round(invoice_amounts.aws_rev_refund, 2) as AWSRefFeeRefund,
round(invoice_amounts.seller_rev_credit, 2) as SellerRevCredit,
round(invoice_amounts.aws_rev_credit, 2) as AWSRefFeeCredit,
(
    round(invoice_amounts.seller_rev_share, 2) +
    round(invoice_amounts.aws_rev_share, 2) +
    round(invoice_amounts.seller_rev_refund, 2) +
    round(invoice_amounts.aws_rev_refund, 2) +
    round(invoice_amounts.seller_rev_credit, 2) +
    round(invoice_amounts.aws_rev_credit, 2)
) as NetAmount,
invoice_amounts.transaction_reference_id
    as TransactionReferenceID,
round(invoice_amounts.seller_tax_share, 2)
    as SellerSalesTax,
round(invoice_amounts.seller_tax_refund, 2)
    as SellerSalesTaxRefund,
payer_info.aws_account_id
    as PayerAwsAccountId,
customer_info.aws_account_id
    as EndCustomerAwsAccountId,
invoice_amounts.disbursement_date
    as DisbursementDate,
invoice_amounts.bank_trace_id
    as BankTraceId
from
    invoice_amounts_aggregated invoice_amounts
join products_with_latest_version products
    on products.product_id = invoice_amounts.product_id
left join accounts_with_latest_version payer_info
    on payer_info.account_id = invoice_amounts.payer_account_id
left join accounts_with_latest_version customer_info
    on customer_info.account_id = invoice_amounts.customer_account_id
)
select * from disbursed_amount_by_product;
```

Laporan kompensasi penjualan

Untuk menemukan pendapatan yang ditagih oleh pelanggan, Anda dapat menjalankan serangkaian kueri seperti berikut ini. Contoh ini sebanding dengan laporan [Laporan kompensasi penjualan](#) penjual. Contoh kueri ini dibangun di atas satu sama lain untuk membuat daftar akhir detail pelanggan dengan jumlah total yang ditagihkan kepada setiap pelanggan untuk penggunaan perangkat lunak Anda. Anda dapat menggunakan kueri seperti yang ditunjukkan, atau menyesuaikannya untuk data dan kasus penggunaan Anda.

Komentar dalam kueri menjelaskan apa yang dilakukan kueri, dan bagaimana memodifikasinya.

Note

Saat menjalankan kueri ini, kami mengasumsikan bahwa data yang dicerna menggunakan dua sumbu waktu (kolom `valid_from` dan `update`). Lihat informasi yang lebih lengkap di [Penyimpanan dan struktur umpan AWS Marketplace data](#).

```
-- Gets all the products and keeps the latest product_id,
-- valid_from tuple.
with products_with_uni_temporal_data as (
  select
    *
  from
    (
      select
        *,
        ROW_NUMBER() OVER (PARTITION BY product_id, valid_from
                           ORDER BY from_iso8601_timestamp(update_date) desc)
        as row_num
      from
        productfeed_v1
    )
  where
    -- A product_id can appear multiple times with the same
    -- valid_from date but with a different update_date column,
    -- making it effectively bi-temporal. By only taking the most
    -- recent tuple, we are converting to a uni-temporal model.
    row_num = 1
),
```

```
-- Gets the latest revision of a product
-- A product can have multiple revisions where some of the
-- columns, like the title, can change.
-- For the purpose of the sales compensation report, we want
-- to get the latest revision of a product
products_with_latest_revision as (
  select
    *
  from
    (
      select
        *,
        ROW_NUMBER() OVER (PARTITION BY product_id ORDER BY
from_iso8601_timestamp(valid_from) desc) as row_num_latest_revision
      from
        products_with_uni_temporal_data
    )
  where
    row_num_latest_revision = 1
),

-- Gets all the addresses and keeps the latest address_id,
-- aws_account_id, and valid_from combination.
-- We're transitioning from a bi-temporal data model to an
-- uni-temporal data_model
piifield_with_uni_temporal_data as (
  select
    *
  from
    (
      select
        *,
        ROW_NUMBER() OVER (
          PARTITION BY address_id, aws_account_id, valid_from
          ORDER BY from_iso8601_timestamp(update_date) desc)
          as row_num
      from
        piifield
    )
  where
    -- An address_id can appear multiple times with the same
    -- valid_from date but with a different update_date column.
    -- We are only interested in the most recent.
    row_num = 1
```

```
),

-- Gets the latest revision of an address.
-- An address_id can have multiple revisions where some of
-- the columns can change.
-- For the purpose of the sales compensation report, we want to
-- get the latest revision of an address + account_id pair.
pii_with_latest_revision as (
  select
    *
  from
    (
      select
        *,
        ROW_NUMBER() OVER (PARTITION BY address_id, aws_account_id
          ORDER BY from_iso8601_timestamp(valid_from) desc)
          as row_num_latest_revision
      from
        piifeed_with_uni_temporal_data
    )
  where
    row_num_latest_revision = 1
),

-- Gets all the accounts and keeps the latest
-- account_id, valid_from tuple.
-- We're transitioning from a bi-temporal data
-- model to an uni-temporal data_model.
accounts_with_uni_temporal_data as (
  select
    *
  from
    (
      select
        *,
        ROW_NUMBER() OVER (PARTITION BY account_id, valid_from
          ORDER BY from_iso8601_timestamp(update_date) desc)
          as row_num
      from
        accountfeed_v1
    )
  where
    -- An account_id can appear multiple times with the same
    -- valid_from date but with a different update_date column.
```

```

    -- We are only interested in the most recent tuple.
    row_num = 1
),

-- Gets all the historical dates for an account
-- An account can have multiple revisions where some of the
-- columns like the mailing_address_id can change.
accounts_with_history as (
  select
    *,
    -- This interval's begin_date
    case
      when
        -- First record for a given account_id
        lag(valid_from, 1) over (partition by account_id
          order by from_iso8601_timestamp(valid_from) asc) is null
      then
        -- 'force' begin_date a bit earlier because of different
        -- data propagation times. We'll subtract one day as one
        -- hour is not sufficient
        from_iso8601_timestamp(valid_from) - INTERVAL '1' DAY
      else
        -- not the first line -> return the real date
        from_iso8601_timestamp(valid_from)
    end as begin_date,
    -- This interval's end date.
    COALESCE(
      LEAD(from_iso8601_timestamp(valid_from), 1)
        OVER (partition by account_id
          ORDER BY from_iso8601_timestamp(valid_from)),
      from_iso8601_timestamp('9999-01-01T00:00:00Z')
    ) as end_date
  from
    accounts_with_uni_temporal_data
),

-- Gets all the billing events and keeps the latest
-- billing_event_id, valid_from tuple.
-- We're transitioning from a bi-temporal data
-- model to an uni-temporal data_model.
billing_events_with_uni_temporal_data as (
  select
    *
  from (

```

```

select
  billing_event_id,
  from_iso8601_timestamp(valid_from) as valid_from,
  from_iso8601_timestamp(update_date) as update_date,
  from_iso8601_timestamp(invoice_date) as invoice_date,
  transaction_type,
  transaction_reference_id,
  product_id,
  disbursement_billing_event_id,
  action,
  currency,
  from_account_id,
  to_account_id,
  end_user_account_id,
  -- convert an empty billing address to null. This will
  -- later be used in a COALESCE call
  case
    when billing_address_id <> '' then billing_address_id else null
  end as billing_address_id,
  CAST(amount as decimal(20, 10)) invoice_amount,
  ROW_NUMBER() OVER (PARTITION BY billing_event_id, valid_from
    ORDER BY from_iso8601_timestamp(update_date) desc)
    as row_num
from
  billingeventfeed_v1
where
  -- The Sales Compensation Report does not contain BALANCE
  -- ADJUSTMENTS, so we filter them out here
  transaction_type <> 'BALANCE_ADJUSTMENT'
  -- Keep only the transactions that will affect any
  -- future disbursed amounts.
  and balance_impacting = '1'
)
where row_num = 1
),

-- Gets the billing address for all DISBURSED invoices. This
-- will be the address of the payer when the invoice was paid.
-- NOTE: For legal reasons, for CPP0 transactions, the
-- manufacturer will not see the payer's billing address id
billing_addresses_for_disbursed_invoices as (
  select
    billing_events_raw.transaction_reference_id,
    billing_events_raw.billing_address_id,

```

```
        billing_events_raw.from_account_id
    from
        billing_events_with_uni_temporal_data billing_events_raw
    where
        -- the disbursed items will contain the billing address id
        billing_events_raw.action = 'DISBURSED'
        -- we only want to get the billing address id for the
        -- transaction line items where the seller is the receiver
        -- of the amount
        and billing_events_raw.transaction_type like 'SELLER_%'
    group by
        billing_events_raw.transaction_reference_id,
        billing_events_raw.billing_address_id,
        billing_events_raw.from_account_id
    ),

-- An invoice can contain multiple line items.
-- We create a pivot table to calculate the different amounts
-- that are part of an invoice.
-- The new row is aggregated at
-- transaction_reference_id - end_user_account_id level
invoiced_and_forgiven_transactions as (
    select
        transaction_reference_id,
        product_id,
        -- A transaction will have the same invoice date for all
        -- of its line items (transaction types)
        max(invoice_date) as invoice_date,
        -- A transaction will have the same billing_address_id
        -- for all of its line items. Remember that the billing event
        -- is uni temporal and we retrieved only the latest valid_from item
        max(billing_address_id) as billing_address_id,
        -- A transaction will have the same currency for all
        -- of its line items
        max(currency) as currency,
        -- We're building a pivot table in order to provide all the
        -- data related to a transaction in a single row
        sum(case when transaction_type = 'SELLER_REV_SHARE'
            then invoice_amount else 0 end) as seller_rev_share,
        sum(case when transaction_type = 'AWS_REV_SHARE'
            then invoice_amount else 0 end) as aws_rev_share,
        sum(case when transaction_type = 'SELLER_REV_SHARE_REFUND'
            then invoice_amount else 0 end) as seller_rev_refund,
        sum(case when transaction_type = 'AWS_REV_SHARE_REFUND'
```

```

        then invoice_amount else 0 end) as aws_rev_refund,
sum(case when transaction_type = 'SELLER_REV_SHARE_CREDIT'
        then invoice_amount else 0 end) as seller_rev_credit,
sum(case when transaction_type = 'AWS_REV_SHARE_CREDIT'
        then invoice_amount else 0 end) as aws_rev_credit,
sum(case when transaction_type = 'SELLER_TAX_SHARE'
        then invoice_amount else 0 end) as seller_tax_share,
sum(case when transaction_type = 'SELLER_TAX_SHARE_REFUND'
        then invoice_amount else 0 end) as seller_tax_refund,
-- this is the account that pays the invoice.
max(case
    -- Get the payer of the invoice from any transaction type
    -- that is not AWS and not BALANCE_ADJUSTMENT.
    -- For AWS and BALANCE_ADJUSTMENT, the billing event feed
    -- will show the "AWS Marketplace" account as the
    -- receiver of the funds and the seller as the payer. We
    -- are not interested in this information here.
    when
        transaction_type not like '%AWS%'
        and transaction_type not like 'BALANCE_ADJUSTMENT'
        then from_account_id
    end) as payer_account_id,
-- this is the account that subscribed to your product
end_user_account_id as customer_account_id
from
    billing_events_with_uni_temporal_data
where
    -- Get invoiced or forgiven items. Disbursements are
    -- not part of the sales compensation report
    action in ('INVOICED', 'FORGIVEN')
group by
    transaction_reference_id,
    product_id,
    -- There might be a different end-user for the same
    -- transaction reference id. Distributed licenses
    -- is an example.
    end_user_account_id
),

invoiced_items_with_product_and_billing_address as (
    select
        invoice_amounts.*,
        products.product_code,
        products.title,

```

```

payer_info.aws_account_id as payer_aws_account_id,
payer_info.account_id as payer_reference_id,
customer_info.aws_account_id as end_user_aws_account_id,
(
    invoice_amounts.seller_rev_share +
    invoice_amounts.aws_rev_share +
    invoice_amounts.seller_rev_refund +
    invoice_amounts.aws_rev_refund +
    invoice_amounts.seller_rev_credit +
    invoice_amounts.aws_rev_credit +
    invoice_amounts.seller_tax_share +
    invoice_amounts.seller_tax_refund
) as seller_net_revenue,
-- Try to get the billing address from the DISBURSED event
-- (if any). If there is no DISBURSEMENT, get the billing
-- address from the INVOICED item. If still no billing address,
-- then default to getting the mailing address of the payer.
coalesce(billing_add.billing_address_id,
         invoice_amounts.billing_address_id,
         payer_info.mailing_address_id)
         as final_billing_address_id
from
    invoiced_and_forgiven_transactions invoice_amounts
join products_with_latest_revision products
    on products.product_id = invoice_amounts.product_id
left join accounts_with_history payer_info
    on payer_info.account_id = invoice_amounts.payer_account_id
    -- Get the Payer Information at the time of invoice creation
    and payer_info.begin_date <= invoice_amounts.invoice_date
    and invoice_amounts.invoice_date < payer_info.end_date
left join accounts_with_history customer_info
    on customer_info.account_id = invoice_amounts.customer_account_id
    -- Get the End User Information at the time of invoice creation
    and customer_info.begin_date <= invoice_amounts.invoice_date
    and invoice_amounts.invoice_date < customer_info.end_date
left join billing_addresses_for_disbursed_invoices billing_add
    on billing_add.transaction_reference_id =
        invoice_amounts.transaction_reference_id
    and billing_add.from_account_id =
        invoice_amounts.payer_account_id
),

invoices_with_full_address as (
select

```

```
payer_aws_account_id as "Customer AWS Account Number",
pii_data.country as "Country",
pii_data.state_or_region as "State",
pii_data.city as "City",
pii_data.postal_code as "Zip Code",
pii_data.email_domain as "Email Domain",
product_code as "Product Code",
title as "Product Title",
seller_rev_share as "Gross Revenue",
aws_rev_share as "AWS Revenue Share",
seller_rev_refund as "Gross Refunds",
aws_rev_refund as "AWS Refunds Share",
seller_net_revenue as "Net Revenue",
currency as "Currency",
date_format(invoice_date, '%Y-%m')as "AR Period",
transaction_reference_id as "Transaction Reference ID",
payer_reference_id as "Payer Reference ID",
end_user_aws_account_id as "End Customer AWS Account ID"
from
invoiced_items_with_product_and_billing_address invoice_amounts
left join pii_with_latest_revision pii_data
    on pii_data.aws_account_id = invoice_amounts.payer_aws_account_id
    and pii_data.address_id = invoice_amounts.final_billing_address_id
-- Filter out FORGIVEN and Field Demonstration Pricing transactions
where seller_net_revenue <> 0
)
select * from invoices_with_full_address;
```

Umpan data

AWS Marketplace menyediakan sejumlah umpan data untuk membantu penjual mengumpulkan dan menganalisis informasi tentang penjualan produk Anda. Umpan data tersedia untuk semua AWS Marketplace penjual terdaftar. Karena umpan data dihasilkan dalam satu hari, mereka berisi data terbaru yang tersedia.

Bagian berikut menjelaskan setiap umpan data.

Topik

- [Umpan data akun](#)
- [Alamat umpan data](#)

- [Umpan data perjanjian](#)
- [Umpan data acara penagihan](#)
- [Umpan data pemetaan lama](#)
- [Tawarkan umpan data](#)
- [Tawarkan umpan data produk](#)
- [Tawarkan umpan data target](#)
- [Umpan data produk](#)
- [Umpan data item pajak](#)

Umpan data akun

Umpan data ini memberikan informasi tentang semua akun yang berinteraksi dengan Anda: akun Anda sendiri, mitra saluran apa pun yang bekerja dengan Anda, pembeli, pembayar, dan semua akun yang dikenakan pajak.

Data akun tidak dapat diubah, dan tidak terkait dengan nomor versi. Perubahan pada bidang ditambahkan, sehingga umpan data ini mungkin memiliki beberapa baris dengan `valid_from` nilai yang sama `account_id` dan berbeda. Untuk informasi tentang bidang riwayat data, lihat [Historisasi data](#).

Umpan data akun disegarkan setiap 24 jam, sehingga data baru tersedia setiap hari.

Tabel berikut menjelaskan nama dan deskripsi kolom umpan data.

Nama kolom	Deskripsi
<code>account_id</code>	<p>Pengidentifikasi unik secara global (GUID) dari akun.</p> <p>Dapat digunakan untuk bergabung ke bidang <code>diProduct</code>, <code>Offer_Target Billing_Event</code>, dan umpan <code>Tax_Item</code> data. Lihat umpan data tersebut untuk informasi tentang bidang yang dapat digunakan untuk bergabung.</p>
<code>aws_account_id</code>	Nomor AWS akun AWS akun penjual, yang unik berdasarkan AWS partisi.

Nama kolom	Deskripsi
encrypted_account_id	ID unik dan terenkripsi untuk pembeli individu aplikasi Anda. Nilai untuk encrypted_account_id digunakan oleh Layanan AWS Marketplace Pengukuran, misalnya, sebagai nilai untuk CustomerIdentifier itu dikembalikan oleh ResolveCustomer tindakan.
mailing_address_id	Referensi alamat surat untuk akun ini.
tax_address_id	Referensi alamat pajak untuk akun ini.
tax_registration_number	Untuk akun non-AS, nomor registrasi pajak untuk akun ini.
tax_legal_name	Untuk akun non-AS, nama perusahaan yang sah. Ini adalah nama yang digunakan pada faktur pajak.

Contoh umpan data akun

Berikut ini menunjukkan contoh feed data akun. Untuk keterbacaan, kolom riwayat data tidak ditampilkan. Untuk informasi tentang bidang riwayat data, lihat [Historisasi data](#).

account_id	aws_account_id	encrypted_account_id	mailing_address_id	tax_address_id	tax_registration_number	tax_legal_name
xk0 CSmiAm6PC 4QqEog9ii aochlzuPI kMfba7a1o DIZ	444456660 000	Zf7 oMzhe GWpH	25o3k46en 6 fiiqtxwx8 e3 3 eViOf kaOiPal UiofjyFa			

account_id	aws_account_id	encrypted_account_id	mailing_address_id	tax_address_id	tax_registration_number	tax_legal_name
7nyo5 Plyx81vx9 ji04 O1ff8 88W8 jwTRo eEwTur biQi	555567679 999	373 Q8v vuQUqm	5oJ6 MRRF2GVH: vj9hfqim8 00m vTjSz uLEHmy FY5Lr42s8	5oJ6 MRRF2GVH: vj9hfqim8 00m vTjSz uLEHmy FY5Lr42s8	SE8239350 83345	
VleGa2t9j 3MuxioH9w c8lsndXXC gGCGUreeX riocM5	737399988 88	8 SPxAYmi8M wX	NLUc5UeiM IGFTrDWC0 ftDPhDUF1 oaSd8xgl5 QM8Db7	V5N hBYBi Yogwy0 WMhndGU4/ fMggmuoTC 2j7Pm8ZKK NNyT	DE4695580 25	

Alamat umpan data

Important

AWS Marketplace akan menghentikan PiiFeed, yang dikirimkan menggunakan [layanan umpan data pengiriman penjual](#), pada bulan Desember 2023. Gunakan umpan data AddressFeed_V1 untuk kebutuhan data alamat Anda. Jika Anda memiliki pertanyaan atau memerlukan dukungan, hubungi tim [Operasi AWS Marketplace Penjual](#).

Umpan data ini menyediakan informasi kontak untuk semua akun yang berinteraksi dengan Anda: akun Anda sendiri, mitra saluran apa pun yang bekerja dengan Anda, pembeli, pembayar, dan semua akun yang dikenakan pajak. Setiap kali transaksi baru terjadi, alamat pelanggan untuk transaksi dipindai, dan jika tidak ada dalam umpan data Anda, entri baru ditambahkan ke file umpan data Anda.

Data alamat tidak dapat diubah.

Umpan data alamat disegarkan setiap 24 jam, sehingga data baru tersedia setiap hari.

Tabel berikut menjelaskan nama dan deskripsi kolom umpan data.

Nama kolom	Deskripsi
address_id	Kunci unik alamat. Dapat digunakan untuk bergabung dari umpan Billing_Event data di billing_address_id lapangan, atau dari umpan Account data pada mailing_address_id atau tax_address_id bidang.
email_domain	Domain untuk alamat email pada file untuk akun ini.
nama_perusahaan	Nama perusahaan pada file untuk akun ini.
country_code	Kode negara ISO 3166 alpha-2 pada file untuk alamat ini.
state_or_region	Negara bagian atau wilayah pada file untuk alamat ini.
kota	Kota pada file untuk alamat ini.
kode_pos	Kode pos pada file untuk alamat ini.
alamat_line_1	Baris pertama dari alamat pada file untuk alamat ini.
alamat_line_2	Baris kedua dari alamat pada file untuk alamat ini.
alamat_line_3	Baris ketiga dari alamat pada file untuk alamat ini.

Contoh umpan data alamat

Berikut ini menunjukkan contoh feed data alamat. Dalam umpan data, informasi ini disajikan dalam satu tabel. Untuk keterbacaan, data ditampilkan dalam dua tabel di sini, dan kolom riwayat data tidak ditampilkan. Untuk informasi tentang bidang riwayat data, lihat [Historisasi data](#).

address_id	email_domain	nama_perusahaan	country_code	state_or_region	kota	kode_pos
V5nhBy 0WMHNDGL TC2j7pm8Z KKnNYT BiYogwy AfMggmuo	a.com	Perusahaan Mateo Jackson	DE		Hamburg	67568
G68xdbkzq dvvhzfbgw 6yf5yos0a 6nisvwhmh 5 ViLjf	b.com	Perusahaan Mary Major	AS	OH	Dayton	57684
Nluc5 UeiMI gftrdwcof tdphduf1o asd8xgl5q m8db7	c.com	Penjual Kami	AS	NY	New York	89475

alamat_line_1	alamat_line_2	alamat_line_3
	Lantai 19	

Umpan data perjanjian

Umpan data ini memberikan informasi tentang perjanjian, yang merupakan kontrak yang ditandatangani antara pengusul (penjual catatan) dan akseptor (AWS Pembeli) untuk mulai menggunakan produk. Umpan data ini memberikan informasi tentang perjanjian untuk penawaran produk yang telah Anda buat sebagai penjual catatan.

Umpan data Perjanjian disegarkan setiap 24 jam.

Tabel berikut mencantumkan dan menjelaskan item dalam umpan data.

Kolom	Deskripsi
valid_dari	Tanggal pertama dimana nilai untuk kunci utama valid dalam kaitannya dengan nilai untuk bidang lain.
insert_date	Tanggal catatan dimasukkan ke dalam umpan data.
update_date	Tanggal rekor terakhir diperbarui.
delete_date	Kolom ini selalu kosong.
Status	Status perjanjian saat ini. Status yang didukung meliputi: <ul style="list-style-type: none">• Aktif — Beberapa atau semua ketentuan perjanjian berlaku.• Diakhiri - Perjanjian berakhir sebelum tanggal akhir yang disepakati sebelumnya karena peristiwa penghentian yang diprakarsai AWS. Paling sering kegagalan pembayaran.• Dibatalkan — Akseptor memilih untuk mengakhiri perjanjian sebelum tanggal berakhirnya.

Kolom	Deskripsi
	<ul style="list-style-type: none"> • Kedaluwarsa - Perjanjian berakhir pada tanggal akhir yang telah disepakati sebelumnya. • Diganti - Perjanjian diganti menggunakan penawaran pengganti. • Diperpanjang - Perjanjian diperbarui menjadi perjanjian baru menggunakan fungsionalitas seperti perpanjangan otomatis. • Diarsipkan — Perjanjian telah berakhir; Alasan pasti untuk berakhirnya, Dihentikan, Dibatalkan, atau Kedaluwarsa, tidak ditentukan. • Rolled back — Revisi perjanjian telah dibatalkan karena kesalahan. Revisi sebelumnya sekarang aktif. Hanya berlaku untuk revisi perjanjian yang tidak aktif. • Superseded — Revisi perjanjian tidak lagi aktif dan revisi perjanjian lainnya sekarang aktif. Hanya berlaku untuk revisi perjanjian yang tidak aktif.
estimasi_agreement_value	Jumlah total yang diketahui pelanggan kemungkinan akan membayar di seluruh siklus hidup perjanjian.
currency_code	Mata uang dari jumlah total yang diketahui pelanggan kemungkinan akan membayar di seluruh siklus hidup perjanjian.
agreement_id	Pengidentifikasi unik perjanjian.
proposer_account_id	Penjual yang mengusulkan ini Purchase Agreement, diwakili oleh pengidentifikasi unik global (GUID) dari akun penjual. Dapat digunakan untuk bergabung ke feed data Akun.

Kolom	Deskripsi
akseptor_account_id	Pembeli yang menerima ini Purchase Agreement, diwakili oleh pengidentifikasi unik global (GUID) dari akun pembeli. Dapat digunakan untuk bergabung ke feed data Akun.
offer_revision_at_acceptance	ID ramah dari penawaran yang sesuai dengan perjanjian ini. Dapat digunakan untuk bergabung ke feed data target Penawaran dan Penawaran.
start_time	Tanggal dan waktu ketika perjanjian dimulai.
waktu_akhir	Tanggal dan waktu ketika perjanjian berakhir. Bidang ini nol untuk pay-as-you-go perjanjian, yang tidak memiliki tanggal akhir.
acceptance_time	Tanggal dan waktu penawaran diterima atau perjanjian dibuat. <ul style="list-style-type: none"> • Dapat di-back-date untuk perjanjian bring-your-own-license • Bisa berbeda dari start_date jika perjanjian dibuat dengan fitur Future-Dated Agreements
maksud	Maksud pembeli ketika perjanjian terakhir diubah.
preceding_agreement_id	ID perjanjian dari perjanjian sebelumnya.
status_reason_code	Alasan perubahan status perjanjian.

Contoh umpan data perjanjian

billing_event_id	seller_account_id	product_id	offer_id	offer_acceptance	status	quantity	unit_price	total_price	merchant_id	pre_agreement_id	status	status	estimated_value	currency
2021-07-20 05:44:17	88axxx	88axxx	tawak	1	2021-07-20 00:00:00	2021-07-20 00:00:00	2021-07-20 00:00:00	BAI		AK			1.00	USD

Umpan data acara penagihan

Umpan data ini memberikan informasi tentang peristiwa penagihan, termasuk faktur dan pencairan.

Misalnya, Anda dapat menggunakan umpan data ini untuk mempelajari kapan dan apa yang ditagih oleh pembeli. Anda juga dapat menggunakan [contoh query SQL](#) untuk menganalisis data dari feed data ini.

Umpan data ini berisi informasi yang terkait dengan peristiwa penagihan di mana Anda adalah penjual catatan. Untuk perjanjian yang dibuat melalui mitra saluran, umpan data ini berisi informasi tentang peristiwa penagihan antara produsen dan penjual rekaman.

Umpan data acara penagihan diperbarui setiap 24 jam, sehingga data baru tersedia setiap hari.

Data acara penagihan tidak dapat diubah.

Tabel berikut menjelaskan nama dan deskripsi kolom umpan data.

Nama kolom	Deskripsi
billing_event_id	Pengenal untuk acara penagihan. ID ini unik di lingkungan penjual.
dari_account_id	Akun yang memulai acara penagihan. Jika transaction_type yaSELLER_RE

Nama kolom	Deskripsi
	<p>V_SHARE , itu adalah akun pembayar pembeli. Ini adalah kunci asing untuk umpan data akun.</p> <p>Dapat digunakan untuk bergabung ke umpan Account data di account_id lapangan.</p>
to_account_id	<p>Akun yang menerima jumlah transaksi untuk produk. Ini adalah kunci asing untuk umpan data akun.</p> <p>Dapat digunakan untuk bergabung ke umpan Account data di account_id lapangan.</p>
end_user_account_id	<p>Akun yang menggunakan produk. Akun ini mungkin berbeda dari akun pembeli dan pembayar.</p> <p>Dapat digunakan untuk bergabung ke umpan Account data di account_id lapangan.</p>
product_id	<p>Pengidentifikasi produk. Ini adalah kunci asing untuk umpan data produk.</p> <p>Dapat digunakan untuk bergabung ke umpan Product data di product_id lapangan.</p>
tindakan	<p>Jenis tindakan untuk acara ini. Kemungkinan nilainya adalah sebagai berikut:</p> <ul style="list-style-type: none"> • INVOICED- Pembeli ditagih untuk jumlah tersebut. • FORGIVEN— Pembeli ditagih untuk jumlah tersebut, dan AWS mengembalikan tagihan. • DISBURSED — Penjual dibayar jumlah ini. Ini dapat mencakup satu bulan faktur, atau menjadi pencairan sesuai permintaan.

Nama kolom	Deskripsi
transaction_type	<p>Jenis transaksi. Sebagai contoh, lihat Skenario perpajakan. Kemungkinan nilainya adalah sebagai berikut:</p> <ul style="list-style-type: none"> • SELLER_REV_SHARE — Jumlah positif; ini adalah harga yang ditetapkan penjual dalam perjanjian dengan pembeli. • SELLER_TAX_SHARE — Jumlah positif; ini adalah jumlah yang ditambahkan SELLER_REV_SHARE untuk menutupi pajak yang berutang penjual. • AWS_REV_SHARE — Jumlah negatif; ini adalah biaya daftar. • AWS_TAX_SHARE — Jumlah positif; ini adalah jumlah pajak yang AWS dikumpulkan di samping SELLER_REV_SHARE. Jumlah ini tidak mempengaruhi saldo penjual. Jumlah ini tidak dicairkan dan disediakan untuk kesadaran penjual tentang pajak yang ditagih kepada pembeli dan dikirimkan ke pihak berwenang atas nama penjual. • <i>transaction_type</i> _REFUND— Jumlah pengembalian dana yang diminta oleh pembeli. • <i>transaction_type</i> _CREDIT— Jumlah AWS kredit pembeli. • BALANCE_ADJUSTMENT — Penyesuaian yang dilakukan oleh AWS untuk menyelesaikan masalah faktur. • DISBURSEMENT — Jika nilai action is DISBURSED dan nilainya balance_impacting adalah 1, ini adalah jumlah yang dibayarkan kepada penjual. Jika

Nama kolom	Deskripsi
	<p>nilainya action adalah INVOICED, catatan ini meniadakan parent_billing_event_id catatan baik secara penuh atau sebagian. Dalam hal ini, pencairan terkait ditampilkan dan nilainya disbursement_billing_event_id adalah. balance_impacting 0</p> <ul style="list-style-type: none">• DISBURSEMENT_FAILURE — Menegasikan transaksi.

Nama kolom	Deskripsi
parent_billing_event_id	<p>Ketika nilai dari broker_id isAWS_INC, nilai action is DISBURSED atauFORGIVEN, dan nilai transaction_type isDISBURSEMENT , parent_billing_event_id mengacu pada aslinya billing_event_id yang memulai acara penagihan ini. Jika action memiliki nilai lain, bidang ini adalah nol.</p> <p>Ketika nilai dari broker_id isAWS_EUROPE , parent_billing_event_id mengacu pada aslinya billing_event_id yang memulai acara penagihan ini untuk skenario berikut:</p> <ul style="list-style-type: none"> • Nilai action is FORGIVEN atau INVOICED dan nilai dari transaction_type is AWS_REV_SHARE ,AWS_REV_SHARE_REFUND , atauSELLER_REV_SHARE_REFUND . • Nilai action is DISBURSED dan nilai transaction_type is ANY (tidak termasukDISBURSEMENT_FAILURE). • Nilai transaction_type adalahAWS_TAX_SHARE ,, AWS_TAX_SHARE_REFUND SELLER_TAX_SHARE , atau SELLER_TAX_SHARE_REFUND . • Nilai action is DISBURSED dan nilai transaction_type isDISBURSEMENT_FAILURE . <p>Ketika nilai broker_id adalahAWS_EUROPE , parent_billing_event_id mengacu pada asli billing_event_id dari peristiwa</p>

Nama kolom	Deskripsi
	<p>penagihan pencairan yang berhasil sebelumnya untuk skenario berikut:</p> <ul style="list-style-type: none"> • Nilai <code>action</code> is <code>DISBURSED</code> dan nilai <code>transaction_type</code> is <code>DISBURSEMENT_FAILURE</code>. <p>Ketika nilai dari <code>broker_id</code> is <code>AWS_EUROPE</code>, bidang ini adalah null untuk semua skenario yang tersisa.</p>
disbursement_billing_event_id	<p>Pencairan terkait ketika nilai <code>action</code> is <code>DISBURSED</code> dan salah satu dari berikut ini benar:</p> <ul style="list-style-type: none"> • <code>transaction_type</code> like ('SELLER%') • <code>transaction_type</code> like ('AWS%') <p>Dalam semua skenario lainnya, nilai ini adalah nol.</p>
jumlah	Jumlah acara penagihan.
mata uang	Kode mata uang ISO 639.
balance_impacting	<p>Apakah jumlah tersebut diperhitungkan dalam menghitung pencairan penjual. Nilai 0 menunjukkan jumlah ditampilkan untuk tujuan informasi dan tidak berpengaruh pada saldo. Nilai 1 menunjukkan bahwa jumlah ini diperhitungkan dalam menentukan pencairan penjual.</p>
invoice_date	Tanggal faktur dibuat.

Nama kolom	Deskripsi
payment_due_date	Ketika nilainya INVOICED, tanggal jatuh tempo untuk faktur. action
usage_period_start_date	Tanggal mulai untuk periode dalam catatan.
usage_period_end_date	Tanggal akhir untuk periode dalam catatan.
invoice_id	ID AWS faktur.
billing_address_id	<p>Referensi alamat penagihan pembayar di umpan data alamat.</p> <p>Dapat digunakan untuk bergabung ke umpan Address data di address_id lapangan.</p>
transaction_reference_id	<p>Pengidentifikasi yang memungkinkan Anda mereferensikan data silang dari laporan berikut:</p> <ul style="list-style-type: none"> • Laporan pencairan • Laporan pendapatan bulanan yang ditagih • Laporan kompensasi penjualan
bank_trace_id	Untuk transaksi pencairan (transaction_type =is DISBURSEMENT dan action isDISBURSED), trace ID yang ditetapkan oleh bank. Trace ID dapat digunakan untuk berkorelasi dengan laporan yang disediakan bank dari bank penjual.

Nama kolom	Deskripsi
broker_id	<p>Pengidentifikasi entitas bisnis yang memfasilitasi transaksi. Kemungkinan nilainya adalah sebagai berikut:</p> <ul style="list-style-type: none"> • AWS_INC <ul style="list-style-type: none"> — Pengidentifikasi untuk AWS, Inc. (berbasis di Amerika Serikat). • AWS_EUROPE <ul style="list-style-type: none"> — Pengidentifikasi untuk Amazon Web Services EMEA SARL (berbasis di Luksemburg). • AWS_JAPAN • AWS_AUSTRALIA • AWS_KOREA • NULL <ul style="list-style-type: none"> — Transaksi sebelumnya tanpa eksplisit broker_id difasilitasi oleh. AWS_INC
buyer_transaction_reference_id	<p>Pengidentifikasi yang mengelompokkan semua catatan terkait dalam umpan penagihan bersama-sama menggunakan GROUP BY atau fungsi jendela dibangun di SQL. Catatan terkait ini dapat berisi faktur pembeli, faktur penjual, dan pajak pertambahan nilai (VATs) pada biaya listing.</p>

Skenario perpajakan

Model perpajakan yang berlaku untuk negara dan negara bagian pembeli dan penjual menentukan bagaimana pajak dikumpulkan dan dikirimkan. Berikut adalah skenario yang mungkin:

- Pajak dikumpulkan dan dikirim oleh AWS. Dalam kasus ini, `transaction_type` adalah `AWS_TAX_SHARE`.
- Pajak dikumpulkan oleh AWS, dicairkan kepada penjual, dan dikirimkan oleh penjual ke otoritas pajak. Dalam kasus ini, `transaction_type` adalah `SELLER_TAX_SHARE`.
- Pajak tidak dikumpulkan oleh AWS. Penjual harus menghitung pajak dan mengirimkannya ke otoritas pajak. Dalam kasus ini, AWS Marketplace tidak melakukan perhitungan pajak atau menerima informasi pajak. Penjual membayar pajak dari bagian pendapatan.

Contoh umpan data peristiwa penagihan

Bagian ini menunjukkan contoh periode data peristiwa penagihan pada saat faktur dan satu bulan kemudian. Perhatikan hal berikut untuk semua tabel di bagian ini:

- Dalam umpan data, `billing_event_id` nilainya adalah string alfanumerik 40 karakter. Mereka ditampilkan di sini sebagai string dua karakter untuk keterbacaan.
- Dalam umpan data, informasi ini disajikan dalam satu tabel. Untuk keterbacaan, data ditampilkan dalam beberapa tabel di sini, dan semua kolom tidak ditampilkan.

Untuk contoh di bagian ini, asumsikan yang berikut:

- Arnav adalah pembeli.
 - ID akunnya adalah `737399998888`.
 - Dia berlokasi di Prancis, yang tunduk pada undang-undang fasilitator pasar. Untuk informasi selengkapnya, lihat [Bantuan Pajak Layanan Web Amazon](#).
 - Dia membeli `prod-o4grxfafcx` dan ditagih \$120,60 untuk penggunaan bulanannya atas produk itu.
 - Dia membayar faktur dalam sebulan.
- Jane adalah pabrikannya.
 - ID akunnya adalah `111122223333`.
- Paulo adalah penjual rekaman.
 - ID akunnya adalah `777788889999`.
 - Dia tinggal di Kansas, yang tidak tunduk pada undang-undang fasilitator pasar.

Umpan data acara penagihan untuk penjual catatan

Sebagai penjual catatan, Paulo menagih pembeli, Arnav.

Tabel berikut menunjukkan informasi yang relevan dalam umpan data Paulo ketika dia menagih Arnav.

billing_event_id	dari_accoun t_id	to_accoun t_id	end_user_ account_id	product_id	tindakan	transacti on_type
I0	737399998 888	777788889 999	737399998 888	prod-o4gr xfafcxxxx	DITAGIH	SELLER_RE V_SHARE
I1	737399998 888	AWS	737399998 888	prod-o4gr xfafcxxxx	DITAGIH	AWS_TAX_B AGIKAN
I2	777788889 999	111122223 333	737399998 888	prod-o4gr xfafcxxxx	DITAGIH	SELLER_RE V_SHARE
I3	777788889 999	AWS	737399998 888	prod-o4gr xfafcxxxx	DITAGIH	AWS_REV_B AGIKAN

parent_bil ling_event_id	disburse ment_bill ing_event_id	jumlah	mata uang	invoice_date	invoice_id
		100	USD	2018-12-31T 00:00:00 Z	781216640
		20.6	USD	2018-12-31T 00:00:00 Z	781216640
		-80	USD	2018-12-31T 00:04:07 Z	788576665
		-0,2	USD	2018-12-31T 00:04:07 Z	788576665

Tabel berikut menunjukkan informasi yang relevan dalam umpan data Paulo pada akhir bulan, setelah Arnav membayar faktur.

billing_event_id	dari_accoun unt_id	to_accoun t_id	end_user_ account_id	product_id	tindakan	transacti on_type
I10	737399998 888	777788889 999	737399998 888		DICAIRKAN	SELLER_RE V_SHARE
I12	777788889 999	111122223 333	737399998 888		DICAIRKAN	SELLER_RE V_SHARE
I13	777788889 999	AWS	737399998 888	prod-o4gr xfafcxxxx	DICAIRKAN	AWS_REV_B AGIKAN
I14	AWS	777788889 999			DICAIRKAN	PENCAIRAN

parent_bil ling_event_id	disbursem ent_bill ing_event_id	jumlah	mata uang	invoice_date	invoice_id
I0	I14	-100	USD	2018-12-31T 00:00:00 Z	781216640
I2	I14	80	USD	2018-12-31T 00:04:07 Z	788576665
I3	I14	0,2	USD	2018-12-31T 00:04:07 Z	788576665
		19.8	USD		

Umpan data acara penagihan untuk produsen

Tabel berikut menunjukkan informasi yang relevan dalam umpan data Jane ketika Paulo menagih Arnav.

billing_event_id	dari_accoun t_id	to_accoun t_id	end_user_ account_id	product_id	tindakan	transacti on_type
I5	777788889 999	111122223 333		prod-o4gr xfafcxxxx	DITAGIH	SELLER_RE V_SHARE
I6	777788889 999	111122223 333		prod-o4gr xfafcxxxx	DITAGIH	SELLER_TA X_SHARE
I7	111122223 333	AWS		prod-o4gr xfafcxxxx	DITAGIH	AWS_REV_B AGIKAN

parent_bil ling_event_id	disbursem ent_bill ing_event_id	jumlah	mata uang	invoice_date	invoice_id
		73.5		2018-12-31T 00:04:07 Z	788576665
		6.5		2018-12-31T 00:04:07 Z	788576665
		-7,35		2018-12-31T 00:04:07 Z	788576665

Tabel berikut menunjukkan informasi yang relevan dalam umpan data Jane pada akhir bulan, setelah faktur dibayar.

billing_e vent_id	dari_accou nt_id	to_accoun t_id	end_user_ account_id	product_id	tindakan	transacti on_type
I30	777788889 999	111122223 333		prod-o4gr xfafcxxxx	DICAIRKAN	SELLER_RE V_SHARE

billing_event_id	dari_account_id	to_account_id	end_user_account_id	product_id	tindakan	transaction_type
I31	77778888999	11112222333		prod-o4grxfafcxxxx	DICAIRKAN	SELLER_TAX_SHARE
I32	11112222333	AWS		prod-o4grxfafcxxxx	DICAIRKAN	AWS_REV_BAGIKAN
I33	AWS	11112222333			DICAIRKAN	PENCAIRAN

parent_billing_event_id	disbursement_billing_event_id	jumlah	mata uang	invoice_date	invoice_id
I5	I33	-73,5	USD		
I6	I33	-6,5	USD		
I7	I33	7.35	USD		
		72,65	USD		

Kueri contoh

Seperti dijelaskan dalam [Mengumpulkan dan menganalisis data dengan umpan data](#), Anda dapat menggunakan [Athena](#) untuk menjalankan kueri pada data yang dikumpulkan dan disimpan sebagai umpan data di bucket Amazon S3 terkelola. Bagian ini memberikan beberapa contoh cara umum yang mungkin Anda lakukan ini. Semua contoh mengasumsikan bahwa satu mata uang digunakan.

Contoh 1: Jumlah yang ditagih, termasuk pajak

Untuk mengetahui berapa banyak pembeli yang ditagih, termasuk pajak, Anda dapat menjalankan kueri seperti yang ditunjukkan pada contoh berikut.

```
SELECT sum(amount) FROM billing_event
WHERE
```

```

action = 'INVOICED'
AND
(
  (transaction_type in ('SELLER_REV_SHARE', 'SELLER_TAX_SHARE'))
  -- to discard SELLER_REV_SHARE from Manufacturer to Channel Partner, aka cost of
  goods
  AND to_account_id='seller-account-id'
)
OR transaction_type= 'AWS_TAX_SHARE'
);

```

Contoh 2: Jumlah yang ditagih kepada pembeli atas nama penjual

Untuk mengetahui berapa banyak pembeli yang ditagih atas nama penjual, Anda dapat menjalankan kueri seperti yang ditunjukkan pada contoh berikut.

```

SELECT sum(amount) FROM billing_event
WHERE
  action = 'INVOICED'
  AND transaction_type in ('SELLER_REV_SHARE', 'SELLER_TAX_SHARE')
  AND to_account_id='seller-account-id'
;

```

Contoh 3: Jumlah AWS dapat dikumpulkan atas nama penjual

Untuk mengetahui berapa banyak yang AWS dapat dikumpulkan atas nama penjual, dikurangi pengembalian uang, kredit, dan akun yang dimaafkan, Anda dapat menjalankan kueri seperti yang ditunjukkan pada contoh berikut.

```

SELECT sum(amount) FROM billing_event
WHERE
  -- what is invoiced on behalf of SELLER, incl. refunds/ credits and cost of goods
  transaction_type like 'SELLER_%'
  -- FORGIVEN action records will "negate" related INVOICED
  and action in ('INVOICED', 'FORGIVEN')
;

```

Contoh 4: Jumlah penjual dapat mengumpulkan

Untuk mengetahui berapa banyak penjual dapat mengumpulkan, Anda dapat menjalankan kueri seperti yang ditunjukkan pada contoh berikut. Contoh ini menghapus biaya listing dan pajak yang AWS mengumpulkan, dan menambahkan penyesuaian saldo yang luar biasa.

```
SELECT sum(amount) FROM billing_event
WHERE
  (transaction_type like 'SELLER_%' -- what is invoiced on behalf of SELLER
  or transaction_type like 'AWS_REV_%' -- what is owed to AWS
  or transaction_type = 'BALANCE_ADJUSTMENT' -- exceptionnal case
  )
  and action in ('INVOICED','FORGIVEN')
;
```

Anda juga dapat menggunakan kueri berikut untuk mengumpulkan informasi yang sama, seperti yang ditunjukkan pada contoh berikut.

```
SELECT sum(amount) FROM billing_event
WHERE
  balance_impacting = 1
  and action in ('INVOICED','FORGIVEN')
;
```

Contoh berikut menunjukkan informasi yang sama, tetapi dibatasi untuk transaksi 2018 dan mengasumsikan semua pembeli membayar faktur mereka.

```
SELECT sum(amount) FROM billing_event
WHERE
  invoice_date between '2018-01-01' and '2018-12-31'
  and balance_impacting = 1
  and action in ('INVOICED','FORGIVEN')
;
```

Contoh 5: Jumlah pencairan

Untuk mengetahui jumlah yang telah dicairkan, Anda dapat menjalankan kueri seperti yang ditunjukkan pada contoh berikut.

```
select sum(amount) FROM billing_event
WHERE
  action = 'DISBURSED'
  and transaction_type like 'DISBURSEMENT%'
;
```

Contoh 6: Jumlah pencairan yang tertunda

Untuk mengetahui jumlah pembayaran yang tertunda, Anda dapat menjalankan kueri seperti yang ditunjukkan pada contoh berikut. Kueri ini menghapus jumlah yang telah dicairkan.

```
SELECT sum(amount) FROM billing_event targeted
WHERE
  (transaction_type like 'SELLER_%' -- what is invoiced on behalf of SELLER
   or transaction_type like 'AWS_REV_%' -- what is owed to AWS
   or transaction_type = 'BALANCE_ADJUSTMENT' -- exceptionnal case
  )
-- DISBURSEMENT action records will "negate" 'INVOICED'
-- but do not take into account failed disbursements
AND
  (not exists
   (select 1
    from billing_event disbursement
    join billing_event failed_disbursement
    on disbursement.billing_event_id=failed_disbursement.parent_billing_event_id
    where
    disbursement.transaction_type='DISBURSEMENT'
    and failed_disbursement.transaction_type='DISBURSEMENT_FAILURE'
    and targeted.disbursement_billing_event_id=disbursement.billing_event_id
   )
  )
;
```

Cara lain untuk mendapatkan informasi yang sama adalah dengan menjalankan kueri untuk mendapatkan saldo penjual, seperti yang ditunjukkan pada contoh berikut.

```
SELECT sum(amount) FROM billing_event
WHERE
  balance_impacting = 1
;
```

Kueri berikut memperluas contoh kita. Ini membatasi hasil untuk transaksi 2018 dan mengembalikan rincian tambahan tentang transaksi.

```
select sum(residual_amount_per_transaction)
from
  (SELECT
    max(billed_invoices.amount) invoiced_amount,
```

```

sum(nvl(disbursed_invoices.amount,0)) disbursed_amount,
-- Exercise left to the reader:
-- use transaction_type to distinguish listing fee vs seller-owed money
-- still pending collection
max(transaction_type) transaction_type,
max(billed_invoices.amount)
  + sum(nvl(disbursed_invoices.amount,0)) residual_amount_per_transaction
FROM billing_event billed_invoices
-- find related disbursements
left join billing_event disbursed_invoices
  on disbursed_invoices.action='DISBURSED'
  and disbursed_invoices.parent_billing_event_id=billed_invoices.billing_event_id
WHERE
  billed_invoices.invoice_date between '2018-01-01' and '2018-12-31'
  and billed_invoices.transaction_type like 'SELLER_%' -- invoiced on behalf of
SELLER
  and billed_invoices.action in ('INVOICED','FORGIVEN')
-- do not take into account failed disbursements
AND not exists
  (select 1 from billing_event failed_disbursement
    where disbursed_invoices.disbursement_billing_event_id =
failed_disbursement.parent_billing_event_id
  )
GROUP BY billed_invoices.billing_event_id
);

```

Contoh 7: Saldo set faktur

Untuk mempelajari jumlah sekumpulan faktur, Anda dapat menjalankan kueri seperti yang ditunjukkan pada contoh berikut.

```

SELECT invoice_id, sum(amount) FROM billing_event targeted
WHERE
-- invoice_id is only not null for invoiced records AND disbursed records
-- linking them to related disbursement -> no need to filter more precisely
invoice_id in ('XXX','YYY')
-- filter out failed disbursements
AND not exists
  (select 1
    from billing_event disbursement
      join billing_event failed_disbursement
        on disbursement.billing_event_id=failed_disbursement.parent_billing_event_id
    where

```

```

disbursement.transaction_type='DISBURSEMENT'
and failed_disbursement.transaction_type='DISBURSEMENT_FAILURE'
and targeted.disbursement_billing_event_id=disbursement.billing_event_id
)
group by invoice_id;

```

Umpan data pemetaan lama

Umpan data ini mencantumkan cara ID produk dan ID penawaran memetakan ke pengidentifikasi unik global (GUID) lama. GUID lama digunakan dalam laporan lama, dan ID baru digunakan dalam umpan data dan API. AWS Marketplace

Umpan data ini memberikan informasi tentang semua produk yang Anda buat sebagai penjual rekaman dan semua produk yang diizinkan untuk dijual kembali.

Umpan data pemetaan lama diperbarui setiap 24 jam, sehingga data baru tersedia setiap hari.

Tabel berikut menjelaskan nama dan deskripsi kolom umpan data.

Nama kolom	Deskripsi
pemetaan_type	Apakah ini ID produk atau ID penawaran.
legacy_id	ID lama untuk produk atau penawaran ini.
new_id	ID ramah untuk produk atau penawaran ini. ID ini digunakan sebagai kunci utama dan dengan semua tindakan API saat ini.

Contoh umpan data pemetaan lama

Berikut ini menunjukkan contoh umpan data pemetaan lama. Untuk keterbacaan, kolom riwayat data tidak ditampilkan. Untuk informasi tentang bidang riwayat data, lihat [Historisasi data](#).

pemetaan_type	legacy_id	new_id
TAWARAN	8a806c74-dbd6-403e-9362-bb08f417ff37	penawaran-dacpxznflfwin

pemetaan_type	legacy_id	new_id
PRODUK	1368541d-890b-4b6c-9bb9-4a55306ab642	prod-o4grxfafcxxxxy
TAWARAN	558d8382-6b3a-4c75-8345-a627b552f5f1	tawaran-gszhmle5npzip

Tawarkan umpan data

Umpan data penawaran memberikan informasi tentang semua penawaran yang telah Anda buat sebagai penjual rekaman. Jika satu penawaran memiliki beberapa revisi, semua revisi disertakan dalam umpan data.

Saat Anda membuat revisi penawaran dan data di bidang terbuka berubah, catatan baru dibuat di umpan data untuk kunci utama yang sama (`offer_id` plus `offer_revision`). Namun, `valid_from` bidang tersebut memiliki nilai yang berbeda. Untuk informasi selengkapnya tentang kolom riwayat umpan data, lihat [Historisasi data](#).

Umpan data penawaran disegarkan setiap 24 jam, sehingga data baru tersedia setiap hari.

Tabel berikut memberikan nama dan deskripsi kolom umpan data.

Nama kolom	Deskripsi
<code>offer_id</code>	<p>Pengenal ramah untuk penawaran.</p> <p>Dapat digunakan untuk bergabung ke <code>offer_id</code> bidang umpan <code>Offer_Product</code> data.</p>
<code>offer_revision</code>	<p>Revisi penawaran. Bidang ini dan <code>offer_id</code> bidang bergabung untuk membentuk kunci utama.</p> <p>Dengan <code>offer_id</code>, dapat digunakan untuk bergabung ke <code>offer_id</code> dan <code>offer_revision</code> bidang umpan <code>Target_Offer</code> data.</p>

Nama kolom	Deskripsi
name	Nama penawaran yang ditentukan penjual.
expiration_date	Tanggal dan waktu penawaran berakhir.
opportunity_name	Setiap data peluang yang terkait dengan penawaran ini. Jika penawaran terikat pada AWS peluang, bidang ini diisi.
opportunity_description	Informasi deskriptif apa pun yang terkait dengan penawaran ini. Jika penawaran terikat pada AWS peluang, bidang ini diisi.
seller_account_id	Pengidentifikasi unik global (GUID) dari akun penjual. Dapat digunakan untuk bergabung dengan account_id bidang di umpan data akun.
opportunity_id	Pengidentifikasi untuk peluang hanya diisi jika reseller menjual produk Anda. Semua penawaran yang dibuat oleh mitra saluran yang berbeda (atau penjual) memiliki hal yang sama opportunity_id jika produknya sama.

Contoh umpan data penawaran

Berikut ini menunjukkan contoh umpan data penawaran. Untuk keterbacaan, kolom riwayat data tidak ditampilkan. Untuk informasi tentang bidang riwayat data, lihat [Historisasi data](#).

offer_id	offer_revisi	nama	tanggal kedaluwarsa_	opportunity_name	opportunity_description	seller_account_id	kesempatan_id
penawaran - dacpxzmfwin	1	Penawaran Program Kontrak	9999-01-01T00:00:00Z				

offer_id	offer_revisi	nama	tanggal kedaluwarsa_	opportunity_name	opportunity_description	seller_account_id	kesempatan_id
		Perusahaan					
tawaran-gszhmle5npzip	1	Penawaran pribadi yang dibuat oleh penjual	2020-10-31T00:00:00Z				
tawaran-hmzhyle8np hlp	1	Penawaran Program Kontrak Perusahaan	9999-01-01T00:00:00Z				

Tawarkan umpan data produk

Satu penawaran dapat memiliki beberapa produk, dan satu produk dapat dimasukkan dalam penawaran yang berbeda. Umpan data ini mencantumkan informasi tentang hubungan antara penawaran dan produk.

Umpan data ini memberikan informasi tentang semua penawaran produk yang Anda buat sebagai penjual rekaman.

Saat Anda menambahkan atau menghapus produk dari penawaran, Anda membuat revisi penawaran.

Umpan data produk penawaran disegarkan setiap 24 jam, sehingga data baru tersedia setiap hari.

Tabel berikut menjelaskan nama dan deskripsi kolom umpan data. Untuk informasi tentang kolom riwayat umpan data, lihat [Historisasi data](#).

Nama kolom	Deskripsi		
offer_id	<p>Pengidentifikasi ramah dari penawaran ini.</p> <p>Dapat digunakan untuk bergabung ke offer_id bidang umpan Offer data.</p>		
offer_revisi	<p>Menggabungkan dengan offer_id bidang untuk membentuk kunci asing untuk revisi penawaran.</p>		
product_id	<p>Pengidentifikasi produk yang ramah, ini adalah kunci asing untuk produk yang diekspos oleh penawaran ini.</p> <p>Dapat digunakan untuk bergabung ke product_id bidang umpan Product data.</p>		

Contoh umpan data produk Penawaran

Berikut ini menunjukkan contoh umpan data produk Penawaran.

offer_id	offer_revisi	product_id
penawaran-dacpxznflfwin	10	prod-o4grxfafcxxxx
tawaran-gszhmle5npzip	24	prod-o4grxfafcxxxxy

Tawarkan umpan data target

Umpan data ini mencantumkan target revisi penawaran untuk semua penawaran yang Anda buat sebagai penjual rekaman. Jika satu penawaran memiliki beberapa revisi, semua revisi disertakan dalam umpan data.

Saat Anda membuat revisi penawaran dan data di bidang terbuka berubah, catatan baru dibuat di umpan data untuk kunci utama yang sama (`offer_id` plus `offer_revision`), tetapi dengan nilai `valid_from` bidang yang berbeda.

Umpan data target penawaran disegarkan setiap 24 jam, sehingga data baru tersedia setiap hari.

Tabel berikut menjelaskan nama dan deskripsi kolom umpan data.

Nama kolom	Deskripsi
<code>offer_target_id</code>	Kunci utama umpan.
<code>offer_id+offer_revision</code>	<p>Pengidentifikasi dan revisi penawaran. Kedua kolom ini merujuk penawaran yang terkait dengan target ini.</p> <p>Dapat digunakan untuk bergabung ke <code>offer_id</code> dan <code>offer_revision</code> bidang umpan Target data.</p>
<code>target_type</code>	Menunjukkan apakah penerima penawaran <code>BuyerAccounts</code> , yang menunjukkan penawaran pribadi, atau <code>ParticipatingPrograms</code> .
<code>polaritas</code>	Menunjukkan apakah penawaran tersebut dimaksudkan untuk dilakukan ke <code>target_type</code> .

Nama kolom	Deskripsi
	<p>pe . Nilai yang dapat diterima adalah sebagai berikut:</p> <ul style="list-style-type: none"> • <code>PositiveTargeting</code> — Penawaran berlaku untuk <code>initarget_type</code> . • <code>NegativeTargeting</code> — Penawaran tidak berlaku untuk <code>initarget_type</code> .
nilai	String yang mewakili target: baik ID AWS akun atau program yang dapat digunakan dengan penawaran. Misalnya, Kontrak Standar untuk AWS Marketplace (SCMP) , atau Program Demonstrasi AWS Marketplace Lapangan (FDP) .

Contoh umpan data target penawaran

Berikut ini menunjukkan contoh umpan data target penawaran. Untuk keterbacaan, kolom riwayat data tidak ditampilkan. Untuk informasi tentang bidang riwayat data, lihat [Historisasi data](#).

offer_target_id	offer_id	offer_revisi	target_type	polaritas	nilai
925ddc73f6a373b7d5544ea3210610803b600	penawaran-dacpxznffwin	1	ParticipatingPrograms	PositiveTargeting	EnterpriseContract
471ff22ae3165278f1fb960d3e14517bcd601	tawaran-gszhmle5npzip	1	ParticipatingPrograms	PositiveTargeting	FieldDemonstration
511ff22adfj65278f1fb960d3e1	tawaran-gszhmle5npzip	1	ParticipatingPrograms	PositiveTargeting	EnterpriseContract

offer_tar get_id	offer_id	offer_revisi	target_type	polaritas	nilai
4517bcd6e 602					

Umpan data produk

Umpan data ini memberikan informasi tentang semua produk yang Anda buat sebagai penjual rekaman dan semua produk yang diizinkan untuk dijual kembali.

Data produk bisa berubah. Ini berarti bahwa ketika Anda mengubah nilai untuk salah satu bidang berikut, catatan baru dibuat dalam umpan data dengan nilai yang berbeda untuk `valid_from` bidang. Untuk informasi selengkapnya tentang kolom riwayat umpan data, lihat [Historisasi data](#).

Umpan data produk disegarkan setiap 24 jam, sehingga data baru tersedia setiap hari.

Tabel berikut menjelaskan nama dan deskripsi kolom umpan data.

Nama kolom	Deskripsi
<code>product_id</code>	<p>Pengidentifikasi produk yang ramah.</p> <p>Dapat digunakan untuk bergabung ke <code>product_id</code> bidang <code>Account</code>, <code>Billing_Event</code>, dan umpan <code>Offer_Product</code> data.</p>
<code>manufacturer_account_id</code>	<p>Pengidentifikasi pemilik produk. Ini adalah kunci asing untuk umpan data Akun.</p> <p>Dapat digunakan untuk bergabung ke <code>account_id</code> bidang umpan <code>Account</code> data.</p>
<code>product_code</code>	<p>Kode produk hak yang ada digunakan untuk mengukur produk. Nilai ini juga digunakan untuk menggabungkan data dengan laporan, atau untuk referensi apa yang disediakan di AWS Marketplace Metering Service.</p>

Nama kolom	Deskripsi
judul	Judul produk.

Contoh umpan data produk

Berikut ini menunjukkan contoh umpan data target penawaran. Untuk keterbacaan, kolom riwayat data tidak ditampilkan. Untuk informasi tentang bidang riwayat data, lihat [Historisasi data](#).

product_id	manufacturer_account_id	product_code	judul
prod-o4grxfafcxxxx	555568000000	product_code_1	Produk1
prod-t3grxfafcxxxy	444457000000	product_code_2	Produk2
prod-x8faxxfafcxy	666678000000	product_code_3	Produk3

Umpan data item pajak

Umpan data ini memberikan informasi tentang perhitungan pajak untuk faktur pelanggan.

Mungkin ada beberapa item baris (`line_item_id`) untuk produk tertentu (`product_id`) dari faktur pelanggan tertentu (`invoice_id`), satu atau lebih untuk setiap yurisdiksi pajak. Ini terjadi, misalnya, dengan tagihan berbasis penggunaan untuk pelanggan yang menggunakan aturan AWS Wilayah yang berbeda oleh AWS entitas yang berbeda (katakanlah, AS dan Irlandia). Untuk mempelajari lebih lanjut tentang tempat AWS mengumpulkan pajak penjualan, PPN, atau GST atas penjualan Anda dan mengirimkan pajak tersebut kepada otoritas pajak setempat, atas nama AWS, Inc., lihat Bantuan Pajak Layanan [Web Amazon](#).

Umpan data item pajak diperbarui setiap 24 jam, sehingga data baru tersedia setiap hari.

Data item pajak tidak dapat diubah.

Tabel berikut menjelaskan nama dan deskripsi kolom umpan data. Untuk informasi tentang kolom riwayat data, lihat [Historisasi data](#).

Nama kolom	Deskripsi
tax_item_id	Pengidentifikasi unik untuk catatan item pajak.
invoice_id	ID AWS faktur. Anda dapat menggunakan nilai ini dengan nilai <code>product_id</code> untuk menemukan acara penagihan pajak terkait.
line_item_id	Pengidentifikasi unik untuk item baris tagihan pelanggan. Transaksi pengembalian dana memiliki ID item baris yang sama dengan transaksi pajak forward mereka.
customer_bill_id	Pengidentifikasi unik tagihan pelanggan. Pembeli dapat membagikan pengenalan ini dengan penjual untuk membantu mengidentifikasi dan menyelesaikan pertanyaan perhitungan pajak.
tax_liable_party	<p>Baik AWS atau Seller. Jika penjual adalah pihak yang bertanggung jawab pajak, pajak dikumpulkan. Jika AWS merupakan pihak yang bertanggung jawab pajak, pajak penjualan dikumpulkan dan dikirimkan oleh AWS. Untuk informasi selengkapnya, lihat AWS Marketplace Penjual & Pengumpulan Pajak.</p> <p>Jika tidak ada pajak yang dikumpulkan, tidak ada nilai yang ditampilkan di sini. Penjual perlu menentukan apakah beberapa pajak dikumpulkan untuk setiap faktur, karena penjual bertanggung jawab atas pengumpulan pajak.</p>
transaction_type_code	<p>Jenis transaksi. Kemungkinan nilainya adalah sebagai berikut:</p> <ul style="list-style-type: none"> • AWS— Transaksi pajak forward

Nama kolom	Deskripsi
	<ul style="list-style-type: none"> • REFUND— Pengembalian dana penuh atau sebagian • TAXONLYREFUND — Pengembalian uang hanya pajak <p>Transaksi pengembalian dana membagikan ID item baris dengan transaksi forward asli mereka.</p>
product_id	<p>Kunci asing untuk produk.</p> <p>Dapat digunakan untuk bergabung ke umpan Product data di product_id lapangan.</p>
product_tax_code	<p>Kode standar untuk mengidentifikasi properti pajak untuk suatu produk. Penjual memilih properti saat membuat atau memodifikasi produk.</p>
invoice_date	<p>Tanggal faktur dibuat.</p>
taxed_customer_account_id	<p>Kunci asing untuk entitas akun yang dikenakan pajak.</p> <p>Dapat digunakan untuk bergabung ke umpan Account data di account_id lapangan.</p>
taxed_customer_country	<p>Kode negara ISO 3166 alpha 2 dari alamat yang digunakan untuk perhitungan pajak.</p>
taxed_customer_state_or_region	<p>Negara bagian, wilayah, atau provinsi yang digunakan untuk perhitungan pajak.</p>
taxed_customer_city	<p>Kota yang digunakan untuk perhitungan pajak.</p>
taxed_customer_postal_code	<p>Kode pos yang digunakan untuk perhitungan pajak.</p>

Nama kolom	Deskripsi
tax_location_code_taxed_yurisdiksi	Geocode simpul yang dikaitkan dengan lokasi yang dikenakan pajak.
tax_type_code	Jenis pajak yang diterapkan pada transaksi. Nilai yang mungkin adalah None, Sales, dan SellerUse .
yurisdiksi_level	Tingkat yurisdiksi alamat yang digunakan untuk lokasi pajak. Nilai yang mungkin adalah State, County, City, dan District.
taxed_yurisdiksi	Nama yurisdiksi pajak.
display_price_taxability_type	Apakah harga yang dilihat pembeli sudah termasuk atau tidak termasuk pajak. Semua AWS Marketplace penawaran tidak termasuk pajak.
taxable_amount	Jumlah transaksi yang dikenakan pajak, pada tingkat yurisdiksi ini.
nontaxable_amount	Jumlah transaksi yang tidak kena pajak, pada tingkat yurisdiksi ini.
tax_jurisdiction_rate	Tarif pajak yang diterapkan, pada tingkat yurisdiksi ini.
tax_amount	Jumlah pajak yang dibebankan, pada tingkat yurisdiksi ini.
tax_currency	Kode mata uang ISO 4217 alpha 3 untuk jumlah di atas.
tax_calculation_reason_code	Apakah transaksi tersebut kena pajak, tidak kena pajak, dikecualikan, atau dinilai nol, yang diatur oleh tingkat yurisdiksi.

Nama kolom	Deskripsi
date_used_for_tax_calculation	Tanggal yang digunakan untuk menghitung pajak atas transaksi.
customer_exemption_certificate_id	ID sertifikat sertifikat pembebasan.
customer_exemption_certificate_id_domain	Lokasi tempat sertifikat disimpan di sistem Amazon.
customer_exemption_certificate_level	Tingkat yurisdiksi yang memberikan pengecualian.
customer_exemption_code	Kode yang menentukan pengecualian; misalnya, . RESALE
customer_exemption_domain	Sistem Amazon yang digunakan untuk menangkap informasi pembebasan pelanggan, jika tersedia.
transaction_reference_id	Pengidentifikasi yang memungkinkan Anda mereferensikan data silang dari laporan berikut: <ul style="list-style-type: none"> • Laporan pencairan • Laporan pendapatan bulanan yang ditagih • Laporan kompensasi penjualan

Note

Mulai 5 Agustus 2021, pajak Marketplace Facilitator internasional untuk AWS Marketplace penjualan akan memiliki entri dalam umpan data item pajak. Ini berarti bahwa, mulai 5 Agustus 2021, setiap AWS_TAX_SHARE catatan SELLER_TAX_SHARE dalam umpan data peristiwa penagihan diharapkan memiliki catatan yang sesuai di umpan data item pajak.

Contoh umpan data item pajak

Berikut ini menunjukkan contoh umpan data item pajak. Dalam umpan data, informasi ini disajikan dalam satu tabel. Untuk keterbacaan, data ditampilkan dalam beberapa tabel di sini, dan semua kolom tidak ditampilkan.

tax_item_id	invoice_id	line_item_id	customer_bill_id
6p2ni6tu041xagvhby anbgxl3xameha16txj oav_0001	781216640	710000000 00000000000	221000000 0000000000
6p2ni6tu041xagvhby anbgxl3xameha16txj oav_0002	781216640	530000000 00000000000	221000000 0000000000
flr4jobxjzww8czdsr q4noue2uxd56j39wxw 0k7_0001	250816266	764000000 00000000000	572000000000000000
gfkjjobxjzw56jgkrs rqjtk52uxd56j39wg j567d_0002	280336288	764000000 00000000000	572439000 0000000000
wwk1qpvb8ran3geiw8 e3mp6dgs2 qj7wpkuwhgk1_0001	451431024	993000000 00000000000	123000000 0000000000
wwk1qpvb8ran3geiw8 e3mp6dgs2 qj7wpkuwhgk1_0002	451431024	993000000 00000000000	312000000 0000000000
fnohdid8kwgqq9lvii 2k30spn3ftgwihbe8h 75x_0001	229987654	921000000 00000000000	639000000 0000000000

tax_liable_party	transacti on_type_code	product_id	product_t ax_code	invoice_date
Penjual	AWS	prod-o4gr xfafcxxx	AWSMP_SOF TWARE_RA	2018-12-31T 00:00:00 Z
Penjual	AWS	prod-o4gr xfafcxxx	AWSMP_SOF TWARE_RA	2018-12-31T 00:00:00 Z
Penjual	AWS	prod-t3gr xfafcxxx	AWS_REMOT E_ACCESS_ SOFTWARE	2018-08-31T 00:00:00 Z
Penjual	PENGEMBAL IAN DANA	prod-t3gr xfafcxxx	AWS_REMOT E_ACCESS_ SOFTWARE	2018-08-31T 00:00:00 Z
Penjual	AWS	prod-x8fa xxfafcxy	AWS_REMOT E_ACCESS_ SOFTWARE	2018-08-31T 00:00:00 Z
Penjual	TAXONLYRE FUND	prod-x8fa xxfafcxy	AWS_REMOT E_ACCESS_ SOFTWARE	2018-05-31T 00:00:00 Z
AWS	AWS	prod-wghj 8xfafrhgj	AWS_REMOT E_ACCESS_ SOFTWARE	2019-07-31T 00:00:00 Z

taxed_cus tomer_acc ount_id	taxed_cus tomer_country	taxed_cus tomer_sta te_or_region	taxed_cus tomer_city	taxed_cus tomer_pos tal_code
VleXXCgGC GUreega2t 9j3muxioh	AS	GA	MILTON	48573-4839

taxed_cus tomer_acc ount_id	taxed_cus tomer_country	taxed_cus tomer_sta te_or_region	taxed_cus tomer_city	taxed_cus tomer_pos tal_code
9wc8lsnd xRloCm5				
VleXXCgGC GUreega2t 9j3muxioh 9wc8lsnd xRloCm5	AS	GA	MILTON	48573-4839
7nyo5jw plyx81vx9ji04 TRo o1ff8biqu 88w8 eEwTur	AS	NC	DURHAM	27517-4834
7nyo5jw plyx81vx9ji04 TRo o1ff8biqu 88w8 eEwTur	AS	NC	DURHAM	27517-4834
7nyo5jw plyx81vx9ji04 TRo o1ff8biqu 88w8 eEwTur	AS	TX	TIDAK BERLAKU	75844-1235
7nyo5jw plyx81vx9ji04 TRo o1ff8biqu 88w8 eEwTur	AS	TX	HOUSTON	75844-1235
192a04213 13e41f069 b52962ed7 babf716291b688	AS	CT	SURGA BARU	06002-2948

tax_location_code_taxed_jurisdiction	tax_type_code	yurisdiksi_level	taxed_yurisdiction	display_price_taxability_type	taxable_amount	nontaxable_amount
460473664	Penjualan	Negara Bagian	GA	Eksklusif	100	0
66301164	Penjualan	Kabupaten	FULTON	Eksklusif	0	100
692938178	SellerUse	Status	NC	Eksklusif	58.1	523,8
692938178	SellerUse	Status	NC	Eksklusif	-58,1	523,8
356794387	Penjualan	Negara Bagian	TX	Eksklusif	1105.14	0
528887443	Penjualan	Kota	HOUSTON	Eksklusif	-36	0
171248162	Penjualan	Negara Bagian	CT	Eksklusif	0	114,55

tax_jurisdiction_rate	tax_amount	tax_currency	tax_calculation_reason_code	date_used_for_tax_calculation
0,206	20.6	USD	Kena pajak	2018-10-31T00:00:00 Z
0	0	USD	NonTaxable	2018-10-31T00:00:00 Z
0.1	5.8	USD	Kena pajak	2018-07-31T00:00:00 Z
0.1	-5,8	USD	Kena pajak	2018-07-31T00:00:00 Z

tax_jurisdiction_rate	tax_amount	tax_currency	tax_calculation_reason_code	date_used_for_tax_calculation
0,06	66.3	USD	Kena pajak	2018-07-31T00:00:00 Z
0,01	-0,36	USD	NonTaxable	2018-07-31T00:00:00 Z
0	0	USD	Dikecualikan	2019-06-30T00:00:00 Z

Laporan penjual di AWS Marketplace

Important

Pada 30 Agustus 2024, AWS Marketplace akan menghentikan beberapa laporan dan kumpulan data:

- Laporan Penjual dipisahkan koma warisan Marketplace (csv)
- API Layanan Analisis Perdagangan (CAS)
- Pemberitahuan email terkait

Periksa halaman laporan untuk laporan yang terkena dampak, atau tanyakan kepada AWS administrator Anda yang diberi komunikasi pada 30/5/2024 jika Anda memiliki pertanyaan.

AWS Marketplace memberikan laporan yang mencakup informasi tentang penggunaan produk, pembeli, penagihan, dan informasi pembayaran. Laporan tersedia untuk semua AWS Marketplace penjual terdaftar.

Berikut adalah beberapa poin penting tentang pembuatan laporan:

- Laporan dibuat setiap hari, mingguan, atau bulanan, tergantung pada laporannya.
- Laporan dibuat pada pukul 00:00 UTC dan mencakup hingga 24:00 UTC hari sebelumnya.

- Laporan dihasilkan sebagai file.csv.
- Anda dapat mengonfigurasi Amazon SNS untuk memberi tahu Anda saat data dikirimkan ke bucket Amazon S3 terenkripsi. Setelah Anda mengonfigurasi notifikasi, AWS kirimkan notifikasi ke alamat email yang terkait dengan AWS akun yang Anda daftarkan AWS Marketplace.

Untuk informasi tentang cara mengonfigurasi notifikasi, lihat [Memulai Amazon SNS](#) di Panduan Pengembang Layanan Pemberitahuan Sederhana Amazon.

Untuk membatalkan mendapatkan email pemberitahuan, hubungi tim [Operasi AWS Marketplace Penjual](#).

- Untuk mempelajari setiap laporan, Anda dapat mengunduh [contoh laporan](#).

Mengakses laporan

AWS Marketplace menyediakan dua cara untuk mengonfigurasi laporan Anda:

- Menggunakan antarmuka API. Hal [Mengakses data produk dan pelanggan dengan Layanan Analisis AWS Marketplace Perdagangan](#) ini memungkinkan Anda untuk secara otomatis mengakses data dalam laporan Anda melalui antarmuka API. Anda dapat mengotomatiskan pengambilan informasi Anda dan mengunduh sebagian laporan alih-alih seluruh laporan. Layanan mengembalikan data secara asinkron ke file di Amazon Simple Storage Service, bukan langsung seperti API tradisional. Data dikirimkan dalam format yang dapat dibaca mesin sehingga Anda dapat mengimpor atau memasukkan data ke dalam sistem Anda.
- Menggunakan dasbor laporan di [Portal Manajemen AWS Marketplace](#). Dasbor ini menyediakan laporan untuk periode pelaporan sebelumnya.

Anda dapat mengontrol akses ke laporan dengan menggunakan izin AWS Identity and Access Management (IAM).

Laporan AWS Marketplace penjual yang tersedia

Laporan-laporan berikut tersedia di AWS Marketplace:

- [Laporan bisnis harian](#)
- [Laporan pelanggan pelanggan harian](#)
- [Laporan pencairan](#)

- [Laporan pendapatan bulanan yang ditagih](#)
- [Laporan kompensasi penjualan](#)

Laporan bisnis harian

Laporan bisnis harian menyediakan data tentang:

- Bagaimana AWS pelanggan menggunakan produk Anda setiap hari
- Estimasi pendapatan dari penggunaan tersebut.

Anda hanya menerima laporan ini jika informasi yang relevan tersedia.

Jika Anda tidak menerima laporan ini dan berpikir bahwa Anda harus, hubungi tim [Operasi AWS Marketplace Penjual](#). Anda juga dapat menghubungi tim tersebut saat ingin membatalkan notifikasi email untuk laporan tersebut.

Anda dapat mengakses laporan ini di [Portal Manajemen AWS Marketplace](#). Jika Anda terdaftar untuk [the section called “AWS Marketplace Layanan Analisis Perdagangan”](#), Anda juga dapat mengakses laporan Anda menggunakan AWS SDK.

Anda dapat menggunakan pengenal unik untuk setiap pelanggan untuk mengidentifikasi pelanggan dari waktu ke waktu dan di seluruh laporan. Pengenal memungkinkan Anda melacak pola penggunaan pelanggan sehingga Anda dapat memperkirakan pengeluaran pelanggan, mendapatkan wawasan tentang penggunaan uji coba gratis, dan tren penggunaan tahunan.

Jadwal publikasi

Laporan ini diterbitkan setiap hari pada pukul 00:00 UTC dan mencakup mulai pukul 00:00 UTC hingga 23:59 UTC hari sebelumnya. Setiap pengecualian untuk jadwal dicatat pada pengenalan bagian laporan bisnis harian.

Topik

- [Bagian 1: Penggunaan berdasarkan jenis instans](#)
- [Bagian 2: Biaya](#)
- [Bagian 3: Konversi uji coba gratis](#)
- [Bagian 4: Contoh baru](#)
- [Bagian 5: Pelanggan produk baru](#)

- [Bagian 6: Pelanggan produk yang dibatalkan](#)

Bagian 1: Penggunaan berdasarkan jenis instans

Bagian ini mencantumkan data dengan baris untuk setiap jenis instance yang digunakan pelanggan. Misalnya, ketika pelanggan menggunakan produk pada satu jenis instance dan produk yang sama pada jenis instance yang berbeda, laporan tersebut menyertakan baris untuk masing-masing dari dua jenis instance.

Nama kolom	Deskripsi
ID Referensi Pelanggan	<p>Pengenal unik yang bukan ID akun. Ini membantu melacak penggunaan, pendapatan, dan langganan oleh pelanggan.</p> <p>ID referensi pelanggan dijadwalkan untuk penghentian pada 31 Maret 2026. Gunakan Akun AWS ID pelanggan sebagai gantinya.</p>
Nomor AWS Rekening Pelanggan	Akun AWS ID tempat tagihan ditagih.
Status Pengguna	Status alamat penagihan yang terkait dengan akun yang berlangganan produk.
Negara Pengguna	<p>Kode negara dua karakter yang dikaitkan dengan akun yang berlangganan produk.</p> <p>Laporan ini menggunakan standar ISO 3166-1 alpha-2.</p>
Judul Produk	Judul produk.
Kode Produk	Pengidentifikasi unik untuk produk.
Tipe Instans	Jenis instance yang terkait dengan penggunaan produk: misalnya, t2.micro.
Unit Pemakaian	Jumlah unit penggunaan yang digunakan pelanggan selama periode pelaporan.

Nama kolom	Deskripsi
Jenis Unit Pemakaian	Unit pengukuran yang mengukur penggunaan pelanggan. Misalnya, jam atau hari.
Deskripsi Penawaran	Deskripsi untuk penawaran produk. Misalnya, produk ditawarkan untuk penggunaan per jam, penggunaan uji coba gratis, atau penggunaan tahunan.
Perkiraan Pendapatan	Estimasi pendapatan dari penggunaan produk. Penagihan diselesaikan pada akhir bulan.
Mata Uang	Mata uang transaksi. Misalnya, jika transaksi dalam dolar AS, entri adalah USD.
ID Penawaran	Pengidentifikasi untuk penawaran yang ditandatangani pembeli.
Menawarkan Visibilitas	Apakah penawaran itu adalah penawaran kontrak publik, swasta, atau perusahaan.
Negara Pelanggan	Kode negara dua karakter yang dikaitkan dengan akun tempat tagihan ditagih.
Judul Solusi	Nama solusinya.
ID Solusi	Pengidentifikasi unik untuk solusinya.
ID Referensi Pembayar	Pengenal unik yang bukan ID akun. Ini terkait dengan akun tempat biaya ditagih. Ini membantu melacak penggunaan, pendapatan, dan langganan oleh pelanggan di semua laporan AWS Marketplace keuangan.
ID Alamat Pembayar	Pengidentifikasi unik yang mewakili alamat pelanggan.

Bagian 2: Biaya

Bagian ini mencakup transaksi berbasis biaya yang terkait dengan produk: misalnya, tahunan, bulanan, biaya produk kontrak SaaS, dan biaya berlangganan produk data. Data dalam bagian ini mencakup periode 24 jam 72 jam sebelum waktu laporan dibuat. Misalnya, jika laporan dibuat pada 24 Mei, data mencakup periode 24 jam untuk 21 Mei.

Nama kolom	Deskripsi
ID Referensi Pelanggan	<p>Pengenal unik yang bukan ID akun. Ini membantu melacak penggunaan, pendapatan, dan langganan oleh pelanggan.</p> <p>ID referensi pelanggan dijadwalkan untuk penghentian pada 31 Maret 2026. Gunakan Akun AWS ID pelanggan sebagai gantinya.</p>
Nomor AWS Rekening Pelanggan	Akun AWS ID tempat tagihan ditagih.
Status Pengguna	Status alamat penagihan yang terkait dengan akun yang berlangganan produk.
Negara Pengguna	<p>Kode negara dua karakter yang dikaitkan dengan akun yang berlangganan produk. Laporan ini menggunakan standar ISO 3166-1 alpha-2.</p>
Judul Produk	Judul produk.
Kode Produk	Pengidentifikasi unik untuk produk.
Jumlah	Biaya penggunaan. Jika ada pengembalian uang, nilai ini negatif. Jika entri ini untuk kontrak AWS Marketplace SaaS, jumlahnya mewakili biaya untuk dimensi, bukan seluruh kontrak.
Mata Uang	Mata uang transaksi. Misalnya, jika transaksi dalam dolar AS, entri adalah USD.

Nama kolom	Deskripsi
Deskripsi Biaya	Alasan biaya: misalnya, biaya bulanan, biaya tahunan, atau pengembalian dana.
Negara Pelanggan	Kode negara dua karakter yang dikaitkan dengan akun tempat tagihan ditagih. Laporan ini menggunakan standar ISO 3166-1 alpha-2.
Negara Pelanggan	Alamat penagihan menyatakan yang terkait dengan akun tempat tagihan ditagih.
Kota Pelanggan	Kota alamat penagihan yang terkait dengan akun yang dikenakan biaya ditagih.
Kode Pos Pelanggan	Kode pos alamat penagihan yang dikaitkan dengan akun tempat tagihan ditagih.
Domain Email Pelanggan	Domain email yang dikaitkan dengan akun tempat tagihan ditagih. Misalnya, jika alamat emailnya <code>liu-jie@example.com</code> , entri adalah <code>example.com</code> .
Tanggal Mulai	Tanggal mulai untuk kontrak AWS Marketplace SaaS atau langganan produk data.
Tanggal Berakhir	Tanggal akhir untuk kontrak AWS Marketplace SaaS atau langganan produk data.
Kuantitas	Jumlah unit untuk dimensi yang ditentukan kontrak.
Dimensi	Dimensi yang ditentukan kontrak.
Judul Solusi	Nama solusinya.
ID Solusi	Pengidentifikasi unik untuk solusinya.

Nama kolom	Deskripsi
ID Referensi Pembayar	Pengenalan unik yang bukan ID akun. Ini terkait dengan akun tempat biaya ditagih. Ini membantu melacak penggunaan, pendapatan, dan langganan oleh pelanggan di semua laporan AWS Marketplace keuangan.
ID Alamat Pembayar	Pengidentifikasi unik yang mewakili alamat pelanggan.

Bagian 3: Konversi uji coba gratis

Bagian ini mencantumkan data untuk uji coba gratis, konversi dan pembatalan, dan mencakup periode 24 jam sebelumnya.

Nama kolom	Deskripsi
Judul Produk	Judul produk.
Kode Produk	Pengidentifikasi unik yang mewakili produk.
Uji Coba Gratis Baru	Jumlah uji coba gratis baru yang dimulai pada periode pelaporan.
Total Uji Coba Gratis Saat Ini	Jumlah total langganan uji coba gratis aktif.
Uji Coba Gratis yang Dikonversi	Jumlah total langganan yang berpindah dari uji coba gratis ke penggunaan berbayar selama periode pelaporan.
Uji Coba Gratis yang Tidak Dikonversi	Jumlah total langganan yang mengakhiri uji coba gratis dan tidak dikonversi ke penggunaan berbayar.
Judul Solusi	Nama solusinya.
ID Solusi	Pengidentifikasi unik untuk solusinya.

Bagian 4: Contoh baru

Bagian ini mencantumkan data untuk jenis EC2 instans dan instans baru, dan mencakup periode 24 jam sebelumnya.

Nama kolom	Deskripsi
ID Referensi Pelanggan	<p>Pengenal unik yang bukan ID akun. Ini membantu melacak penggunaan, pendapatan, dan langganan oleh pelanggan.</p> <p>ID referensi pelanggan dijadwalkan untuk penghentian pada 31 Maret 2026. Gunakan Akun AWS ID pelanggan sebagai gantinya.</p>
Nomor AWS Rekening Pelanggan	Akun AWS ID tempat tagihan ditagih.
Status Pengguna	Status alamat penagihan yang terkait dengan akun yang berlangganan produk.
Negara Pengguna	Kode negara dua karakter yang dikaitkan dengan akun yang berlangganan produk. Laporan ini menggunakan standar ISO 3166-1 alpha-2.
Judul Produk	Judul produk.
Kode Produk	Pengidentifikasi unik untuk produk.
Tipe	Jenis EC2 instans Amazon.
Hitungan	Jumlah EC2 contoh.
Negara Pelanggan	Kode negara dua karakter yang dikaitkan dengan akun tempat tagihan ditagih. Laporan ini menggunakan standar ISO 3166-1 alpha-2.
Negara Pelanggan	Alamat penagihan menyatakan yang terkait dengan akun tempat tagihan ditagih.

Nama kolom	Deskripsi
Kota Pelanggan	Kota alamat penagihan yang terkait dengan akun yang dikenakan biaya ditagih.
Kode Pos Pelanggan	Kode pos alamat penagihan yang dikaitkan dengan akun tempat tagihan ditagih.
Domain Email Pelanggan	Domain email yang dikaitkan dengan akun tempat tagihan ditagih. Misalnya, jika alamat emailnya <code>liu-jie@example.com</code> , entri adalah <code>example.com</code> .
Judul Solusi	Nama solusinya.
ID Solusi	Pengidentifikasi unik untuk solusinya.
ID Referensi Pembayar	Pengenal unik yang bukan ID akun. Ini terkait dengan akun tempat biaya ditagih. Ini membantu melacak penggunaan, pendapatan, dan langganan oleh pelanggan di semua laporan AWS Marketplace keuangan.
ID Alamat Pembayar	Pengidentifikasi unik yang mewakili alamat pelanggan.

Bagian 5: Pelanggan produk baru

Bagian ini mencantumkan data untuk pembeli baru, dan mencakup periode 24 jam sebelumnya.

Nama kolom	Deskripsi
ID Referensi Pelanggan	Pengenal unik yang bukan ID akun. Ini membantu melacak penggunaan, pendapatan, dan langganan oleh pelanggan.

Nama kolom	Deskripsi
	ID referensi pelanggan dijadwalkan untuk penghentian pada 31 Maret 2026. Gunakan Akun AWS ID pelanggan sebagai gantinya.
Nomor AWS Rekening Pelanggan	Akun AWS ID tempat tagihan ditagih.
Status Pengguna	Status alamat penagihan yang terkait dengan akun yang berlangganan produk.
Negara Pengguna	Kode negara dua karakter yang dikaitkan dengan akun berlangganan produk. Laporan ini menggunakan standar ISO 3166-1 alpha-2.
Judul Produk	Judul produk.
Kode Produk	Pengidentifikasi unik untuk produk.
ID Penawaran	Pengidentifikasi untuk penawaran yang ditandatangani pembeli.
Menawarkan Visibilitas	Apakah penawaran itu adalah penawaran kontrak publik, swasta, atau perusahaan.
Negara Pelanggan	Kode negara dua karakter yang dikaitkan dengan akun tempat tagihan ditagih. Laporan ini menggunakan standar ISO 3166-1 alpha-2.
Negara Pelanggan	Alamat penagihan menyatakan yang terkait dengan akun tempat tagihan ditagih.
Kota Pelanggan	Kota alamat penagihan yang terkait dengan akun yang dikenakan biaya ditagih.
Kode Pos Pelanggan	Kode pos alamat penagihan yang dikaitkan dengan akun tempat tagihan ditagih.

Nama kolom	Deskripsi
Domain Email Pelanggan	Domain email yang dikaitkan dengan akun tempat tagihan ditagih. Misalnya, jika alamat emailnya <code>aliu-jie@example.com</code> , entri adalah <code>example.com</code> .
Judul Solusi	Nama solusinya.
ID Solusi	Pengidentifikasi unik untuk solusinya.
ID Referensi Pembayar	Pengenal unik yang bukan akun. Ini terkait dengan akun tempat biaya ditagih. Ini membantu melacak penggunaan, pendapatan, dan langganan oleh pelanggan di semua laporan AWS Marketplace keuangan.
ID Alamat Pembayar	Pengidentifikasi unik yang mewakili alamat pelanggan.

Bagian 6: Pelanggan produk yang dibatalkan

Bagian ini mencantumkan data untuk pembatalan pembeli, dan mencakup periode 24 jam sebelumnya.

Nama kolom	Deskripsi
ID Referensi Pelanggan	<p>Pengenal unik yang bukan ID akun. Ini membantu melacak penggunaan, pendapatan, dan langganan oleh pelanggan.</p> <p>ID referensi pelanggan dijadwalkan untuk penghentian pada 31 Maret 2026. Gunakan Akun AWS ID pelanggan sebagai gantinya.</p>
Nomor AWS Rekening Pelanggan	Akun AWS ID tempat tagihan ditagih.

Nama kolom	Deskripsi
Status Pengguna	Status alamat penagihan yang terkait dengan akun yang berlangganan produk.
Negara Pengguna	Kode negara dua karakter yang dikaitkan dengan akun yang berlangganan produk. Laporan ini menggunakan standar ISO 3166-1 alpha-2.
Judul Produk	Judul produk.
Kode Produk	Pengidentifikasi unik untuk produk.
Tanggal Berlangganan	Tanggal ketika langganan dimulai.
ID Penawaran	Pengidentifikasi untuk penawaran yang ditandatangani pembeli.
Menawarkan Visibilitas	Apakah penawaran itu adalah penawaran kontrak publik, swasta, atau perusahaan.
Negara Pelanggan	Kode negara dua karakter yang dikaitkan dengan akun tempat tagihan ditagih. Laporan ini menggunakan standar ISO 3166-1 alpha-2.
Negara Pelanggan	Alamat penagihan menyatakan yang terkait dengan akun tempat tagihan ditagih.
Kota Pelanggan	Kota alamat penagihan yang terkait dengan akun yang dikenakan biaya ditagih.
Kode Pos Pelanggan	Kode pos alamat penagihan yang dikaitkan dengan akun tempat tagihan ditagih.
Domain Email Pelanggan	Domain email yang dikaitkan dengan akun tempat tagihan ditagih. Misalnya, jika alamat emailnya <code>liu-jie@example.com</code> , entri adalah <code>example.com</code> .

Nama kolom	Deskripsi
Judul Solusi	Nama solusinya.
ID Solusi	Pengidentifikasi unik untuk solusinya.
ID Referensi Pembayar	Pengenal unik yang bukan ID akun. Ini terkait dengan akun tempat biaya ditagih. Ini membantu melacak penggunaan, pendapatan, dan langganan oleh pelanggan di semua laporan AWS Marketplace keuangan.
ID Alamat Pembayar	Pengidentifikasi unik yang mewakili alamat pelanggan.

Laporan pelanggan pelanggan harian

Laporan ini mencantumkan data untuk pelanggan yang membeli produk Anda. Laporan ini tidak menentukan penggunaan saat ini atau sebelumnya, hanya pelanggan yang berlangganan produk Anda. Anda hanya menerima laporan ini jika informasi yang relevan tersedia. Jika Anda tidak menerima laporan ini dan berpikir bahwa Anda harus, hubungi tim [Operasi AWS Marketplace Penjual](#). Anda juga dapat menghubungi tim Operasi AWS Marketplace Penjual untuk membatalkan pemberitahuan email untuk laporan ini.

Anda dapat mengakses laporan ini di [Portal Manajemen AWS Marketplace](#). Jika Anda terdaftar untuk [the section called “AWS Marketplace Layanan Analisis Perdagangan”](#), Anda juga dapat mengakses laporan Anda menggunakan AWS SDK.

Laporan ini memiliki dua bagian: satu untuk langganan per jam dan bulanan dan satu untuk langganan tahunan. Laporan tersebut mencakup daftar AWS akun IDs untuk semua pelanggan yang berlangganan produk Anda.

Jadwal publikasi

Laporan ini diterbitkan setiap hari pada pukul 00:00 UTC dan mencakup mulai pukul 00:00 UTC hingga 23:59 UTC hari sebelumnya.

Topik

- [Bagian 1: Langganan per jam dan bulanan](#)

- [Bagian 2: Langganan panjang variabel](#)

Bagian 1: Langganan per jam dan bulanan

Bagian ini mencantumkan data untuk semua langganan berbasis penggunaan pada hari sebelumnya pada pukul 23:59:59 UTC.

Nama kolom	Deskripsi
Nomor AWS Rekening Pelanggan	Akun AWS ID tempat tagihan ditagih.
Judul Produk	Judul produk.
Id Produk	Pengidentifikasi unik untuk produk perangkat lunak.
Kode Produk	Pengidentifikasi unik untuk produk perangkat lunak.
Tanggal Mulai langganan	Tanggal mulai untuk berlangganan, diformat sebagaiYYYY-MM-DD .
ID Penawaran	Pengidentifikasi untuk penawaran yang ditandatangani pembeli.
Menawarkan Visibilitas	Apakah penawaran itu adalah penawaran kontrak publik, swasta, atau perusahaan.
Judul Solusi	Nama solusinya.
ID Solusi	Pengidentifikasi unik untuk solusinya.
ID Referensi Pembayar	Pengenal unik yang bukan ID akun. Ini terkait dengan akun tempat biaya ditagih. Ini membantu melacak penggunaan, pendapatan, dan langganan oleh pelanggan di semua laporan AWS Marketplace keuangan.
ID akun pengecer	Pengenal unik untuk reseller mitra saluran.

Nama kolom	Deskripsi
Nama akun reseller	Nama reseller mitra saluran.

Bagian 2: Langganan panjang variabel

Bagian ini mencantumkan data untuk semua langganan berbasis biaya pada hari sebelumnya pada pukul 23:59:59 UTC.

Nama kolom	Deskripsi
Nomor AWS Rekening Pelanggan	Akun AWS ID tempat tagihan ditagih.
Judul Produk	Judul produk.
Id Produk	Pengidentifikasi unik untuk produk perangkat lunak.
Kode Produk	Pengidentifikasi unik untuk produk perangkat lunak. Informasi ini juga tersedia sebagai bagian dari metadata EC2 instans Amazon.
Id Berlangganan	ID untuk berlangganan.
Kuantitas Berlangganan	Jumlah total lisensi yang dibeli pelanggan.
Jenis Berlangganan	Jenis berlangganan.
Niat Berlangganan	Apakah penawaran ini merupakan peningkatan atau pembaruan dari penawaran sebelumnya.
ID Penawaran	Pengidentifikasi untuk penawaran yang ditandatangani pembeli.
Tanggal Mulai langganan	Tanggal ketika pelanggan berlangganan produk, diformat sebagai. YYYY-MM-DD
ID Penawaran Sebelumnya	ID penawaran yang mendahului penawaran peningkatan atau pembaruan, jika ada.

Nama kolom	Deskripsi
Menawarkan Visibilitas	Apakah penawaran itu adalah penawaran kontrak publik, swasta, atau perusahaan.
Judul Solusi	Nama solusinya.
ID Solusi	Pengidentifikasi unik untuk solusinya.
ID Referensi Pembayar	Pengenal unik yang bukan ID akun. Ini terkait dengan akun tempat biaya ditagih. Ini membantu melacak penggunaan, pendapatan, dan langganan oleh pelanggan di semua laporan AWS Marketplace keuangan.
ID akun pengecer	Pengenal unik untuk reseller mitra saluran.
Nama akun reseller	Nama reseller mitra saluran.

Laporan pencairan

Important

Laporan ini tidak akan tersedia pada 30 Agustus 2024. Anda dapat menggunakan alternatif ini:

- [Dasbor Koleksi dan pencairan](#)
- Kueri [SQL collections_and_disbursements_report aktif](#). GitHub

Laporan pencairan memberikan informasi tentang dana yang kami kumpulkan dan dicairkan ke rekening bank Anda sejak pencairan sebelumnya. Pencairan dapat mencakup pembayaran pelanggan atau pengembalian uang untuk berlangganan produk Anda, dan beberapa pajak yang dikumpulkan atau dikembalikan kepada pelanggan. Anda tidak menerima pencairan dana sampai dana dikumpulkan dari nasabah. Pelanggan yang berbeda memiliki ketentuan pembayaran yang berbeda AWS, sehingga sebagian dana di masing-masing kategori usia yang tidak tertagih mungkin tidak jatuh tempo dari pelanggan.

Pengembalian uang muncul sebagai jumlah negatif karena uang dikembalikan ke pelanggan Anda setelah Anda mengotorisasi pengembalian dana.

Laporan ini tersedia di Portal Manajemen AWS Marketplace bawah tab Laporan. Untuk membuat laporan Anda sendiri yang mirip dengan yang satu ini AWS Marketplace [Umpan data pengiriman penjual di AWS Marketplace](#), Anda dapat menggunakan, termasuk [Pencairan berdasarkan produk](#) sebagai laporan dasar untuk menyesuaikan untuk memenuhi kebutuhan Anda.

Note

Untuk membatalkan email pemberitahuan untuk laporan ini, hubungi tim [Operasi AWS Marketplace Penjual](#).

Jadwal publikasi

Laporan ini diterbitkan 3-5 hari setelah pencairan dimulai untuk mentransfer dana ke bank Anda. Secara umum, ini adalah laporan untuk penjual yang menerima pencairan pada irama bulanan. Jika tidak ada pencairan yang dimulai, tidak ada laporan pencairan yang dihasilkan.

Topik

- [Bagian 1: Jumlah yang dicairkan berdasarkan produk](#)
- [Bagian 2: Jumlah yang dicairkan berdasarkan geografi pelanggan](#)
- [Bagian 3: Jumlah yang dicairkan berdasarkan jam instans](#)
- [Bagian 4: Usia dana yang tidak tertagih](#)
- [Bagian 5: Usia dana yang dicairkan](#)
- [Bagian 6: Usia dana yang jatuh tempo](#)
- [Bagian 7: Rincian dana yang tidak dikumpulkan](#)

Bagian 1: Jumlah yang dicairkan berdasarkan produk

Bagian ini mencantumkan data untuk pencairan berdasarkan produk.

Nama kolom	Deskripsi
Produk	Judul produk.

Nama kolom	Deskripsi
Kode Produk	Pengidentifikasi unik untuk produk.
SellerRev	Jumlah yang ditagihkan kepada pelanggan untuk penggunaan atau biaya produk.
AWSRefBiaya	Jumlah AWS Marketplace biaya.
SellerRevRefund	Jumlah biaya berlangganan yang dikembalikan kepada pelanggan jika ada pengembalian uang yang diproses selama periode cakupan data.
AWSRefFeeRefund	Jumlah AWS Marketplace biaya yang dikembalikan jika ada pengembalian uang yang diproses selama periode cakupan data.
SellerRevCredit	AWS Kredit yang AWS Marketplace ditempatkan di akun pelanggan.
AWSRefFeeCredit	AWS Kredit yang AWS Marketplace ditempatkan di akun Anda.
Jumlah Bersih	Total dana yang kami salurkan kepada Anda. Kolom ini sama dengan SellerRev kolom dikurangi kolom AWSRef Biaya. Ketika pengembalian dana diberikan kepada pelanggan, kolom ini adalah angka negatif yang sama dengan SellerRevRefund kolom dikurangi AWSRef FeeRefund kolom.
ID Referensi Transaksi	Pengidentifikasi unik untuk transaksi yang membantu Anda menghubungkan transaksi di seluruh AWS Marketplace laporan.
USSalesPajak Penjual	Jumlah total pajak penjualan dan penggunaan AS yang ditagih untuk transaksi ini.

Nama kolom	Deskripsi
Penjual USSales TaxRefund	Jumlah total pajak penjualan dan penggunaan AS yang dikembalikan untuk transaksi ini jika pengembalian dana diproses.
Nomor AWS Rekening Pelanggan	ID akun tempat tagihan ditagih.
Negara Pelanggan	Kode negara dua karakter yang dikaitkan dengan akun tempat tagihan ditagih. Laporan ini menggunakan standar ISO 3166-1 alpha-2.
Negara Pelanggan	Alamat penagihan menyatakan yang terkait dengan akun tempat tagihan ditagih.
Kota Pelanggan	Kota alamat penagihan yang terkait dengan akun yang dikenakan biaya ditagih.
Kode Pos Pelanggan	Kode pos alamat penagihan yang dikaitkan dengan akun tempat tagihan ditagih.
Domain Email Pelanggan	Domain email yang dikaitkan dengan akun tempat tagihan ditagih. Misalnya, jika alamat emailnya <code>liu-jie@example.com</code> , entri adalah <code>example.com</code> .
Judul Solusi	Nama solusinya.
ID Solusi	Pengidentifikasi unik untuk solusinya.
ID Referensi Pembayar	Pengenal unik yang bukan ID akun. Ini terkait dengan akun tempat biaya ditagih. Ini membantu melacak penggunaan, pendapatan, dan langganan oleh pelanggan di semua laporan AWS Marketplace keuangan.
ID Alamat Pembayar	Pengenal unik yang mewakili alamat pelanggan.

Bagian 2: Jumlah yang dicairkan berdasarkan geografi pelanggan

Bagian ini mencantumkan data untuk pencairan berdasarkan lokasi geografis pelanggan.

Nama kolom	Deskripsi
ID Penyelesaian	Pengidentifikasi unik pencairan.
Tanggal Mulai Periode Penyelesaian	Tanggal mulai dan waktu periode pencairan.
Tanggal Akhir Periode Penyelesaian	Tanggal akhir dan waktu periode pencairan.
Tanggal Setoran	Tanggal dan waktu pencairan terjadi.
Jumlah yang dicairkan	Jumlah total pencairan.
Kode Negara	Kode negara dua karakter yang dikaitkan dengan akun tempat tagihan ditagih. Laporan ini menggunakan standar ISO 3166-1 alpha-2.
Negara atau Wilayah	Alamat penagihan menyatakan yang terkait dengan akun tempat tagihan ditagih.
Kota	Kota alamat penagihan yang terkait dengan akun yang dikenakan biaya ditagih.
Kode Pos	Kode pos alamat penagihan yang dikaitkan dengan akun tempat tagihan perangkat lunak ditagih.
Jumlah Bersih berdasarkan Lokasi Pajak	Total dana yang dicairkan kepada penjual berdasarkan lokasi pajak, lebih sedikit AWS Marketplace biaya, pengembalian uang, dan pajak penjualan dan penggunaan AS.
Jumlah Kotor berdasarkan Lokasi Pajak	Total dana yang disalurkan kepada penjual berdasarkan lokasi pajak.
Penjual Pajak Penjualan AS	Jumlah total pajak penjualan dan penggunaan AS yang ditagih untuk transaksi ini atas nama

Nama kolom	Deskripsi
	Penjual. (Artinya, catatan terkait dalam laporan Penjualan dan Pajak AS menunjukkan “pihak yang bertanggung jawab pajak” == “PENJUAL” .)
Penjual Pengembalian Pajak Penjualan AS	Jumlah total pajak penjualan dan penggunaan AS yang dikembalikan untuk transaksi ini jika pengembalian dana diproses, ketika pajak tersebut dikumpulkan atas nama Penjual. (Artinya, catatan terkait dalam laporan Penjualan dan Pajak AS menunjukkan “pihak yang bertanggung jawab pajak” == “PENJUAL” .)

Bagian 3: Jumlah yang dicairkan berdasarkan jam instans

Bagian ini mencantumkan data untuk pencairan berdasarkan jam instans Amazon EC2 .

Nama kolom	Deskripsi
Produk	Judul produk.
Kode Produk	Pengidentifikasi unik untuk produk.
Deskripsi Jenis Penggunaan	Deskripsi penggunaan, termasuk jenis penawaran, Wilayah, dan jenis instans.
Laju	Tarif per jam untuk jenis penawaran, Wilayah, dan jenis instans.
Hitungan Pengguna	Jumlah pelanggan unik yang menggunakan jenis penawaran, Wilayah, dan jenis instans.
Jam Instance	Jumlah jam yang digunakan instans untuk jenis penawaran, Wilayah, dan jenis instans.

Nama kolom	Deskripsi
Judul Solusi	Nama solusinya.
ID Solusi	Pengidentifikasi unik untuk solusinya.

Bagian 4: Usia dana yang tidak tertagih

Bagian ini mencantumkan data untuk dana yang tidak dikumpulkan, yang diatur berdasarkan usia. Dana yang tidak terkumpul mungkin termasuk jumlah yang belum jatuh tempo.

Nama kolom	Deskripsi
Tidak dikumpulkan (< 31 hari tertunda)	Total dana yang ditagih tetapi tidak dikumpulkan kurang dari 31 hari.
Tidak dikumpulkan (31—60 hari tertunda)	Total dana yang ditagih tetapi tidak dikumpulkan selama antara 31-60 hari.
Tidak dikumpulkan (61—90 hari tertunda)	Total dana yang ditagih tetapi tidak dikumpulkan selama antara 61-90 hari.
Tidak dikumpulkan (91—120 hari tertunda)	Total dana yang ditagih tetapi tidak dikumpulkan selama antara 91-120 hari.
Tidak dikumpulkan (> 120 hari tertunda)	Total dana yang ditagih tetapi tidak dikumpulkan selama lebih dari 120 hari.
Tidak dikumpulkan (keseluruhan)	Total semua dana ditagih tetapi tidak dikumpulkan.

Bagian 5: Usia dana yang dicairkan

Bagian ini mencantumkan data untuk dana yang dikumpulkan sejak pencairan sebelumnya.

Nama kolom	Deskripsi
Dikumpulkan (< 31 hari tertunda)	Total dana yang dikumpulkan yang ditagih dalam kisaran 0-31 hari.
Dikumpulkan (31—60 hari tertunda)	Total dana yang dikumpulkan yang ditagih dalam kisaran 31-60 hari.
Dikumpulkan (61—90 hari tertunda)	Total dana yang dikumpulkan yang ditagih dalam kisaran 61-90 hari.
Dikumpulkan (91—120 hari tertunda)	Total dana yang dikumpulkan yang ditagih dalam kisaran 91-120 hari.
Dikumpulkan (> 120 hari tertunda)	Total dana yang terkumpul yang ditagih dalam rentang lebih dari 120 hari.
Dikumpulkan (keseluruhan)	Total semua dana yang dikumpulkan.

Bagian 6: Usia dana yang jatuh tempo

Bagian ini mencantumkan data untuk dana yang telah diperoleh dan dibayarkan oleh pelanggan, tetapi belum dibayar sesuai dengan perjanjian pelanggan dengan AWS

Nama kolom	Deskripsi
Masa Lalu Jatuh Tempo (< 31 hari)	Total dana yang telah terkumpul dalam 0-31 hari terakhir dan jatuh tempo tetapi pelanggan belum membayar.
Masa Lalu Jatuh Tempo (31—60 hari)	Total dana yang telah terkumpul dalam 31-60 hari terakhir dan jatuh tempo tetapi pelanggan belum membayar.
Masa Lalu Jatuh Tempo (61—90 hari)	Total dana yang telah terkumpul dalam 61-90 hari terakhir yang jatuh tempo tetapi pelanggan belum membayar.

Nama kolom	Deskripsi
Masa Lalu Jatuh Tempo (91—120 hari)	Total dana yang telah terkumpul dalam 91-120 hari terakhir dan jatuh tempo tetapi pelanggan belum membayar.
Masa Lalu Jatuh Tempo (> 120 hari)	Total dana yang telah terkumpul dalam 121 hari terakhir atau lebih dan jatuh tempo tetapi pelanggan belum membayar.
Masa Lalu Jatuh Tempo (keseluruhan)	Total dana yang telah terkumpul dan jatuh tempo tetapi pelanggan belum membayar.

Bagian 7: Rincian dana yang tidak dikumpulkan

Bagian ini mencantumkan semua dana yang tidak dikumpulkan, diurutkan berdasarkan tanggal jatuh tempo pembayaran.

Nama kolom	Deskripsi
Nomor AWS Rekening Pembayar	Akun yang dikenakan biaya perangkat lunak.
Kode Produk	Pengidentifikasi unik untuk produk.
Pendapatan Bruto	Jumlah yang ditagih untuk menggunakan produk atau biaya untuk menggunakan produk.
AWS Bagi Pendapatan	Jumlah AWS biaya yang dikurangkan dari jumlah yang ditagih pada waktu penyelesaian.
Pengembalian Bruto	Jumlah total pengembalian uang untuk transaksi.
AWS Pengembalian Uang Berbagi	Bagian dari AWS biaya yang dikembalikan untuk transaksi.

Nama kolom	Deskripsi
Pendapatan Bersih	Jumlah bersih yang ditagih untuk transaksi ini, dikurangi AWS biaya, pengembalian uang, dan pajak penjualan dan penggunaan AS.
Mata Uang	Mata uang transaksi. Misalnya, jika transaksi dalam dolar AS, entri adalah USD.
Periode AR	Bulan dan tahun transaksi, dalam formatYYYY-MM.
ID Referensi Transaksi	Pengidentifikasi unik yang mewakili transaksi , yang dapat Anda gunakan untuk menghubungkan transaksi di seluruh AWS Marketplace laporan.
Nama Peluang	Pengidentifikasi unik untuk peluang terdaftar.
Deskripsi Peluang	Setiap metadata dalam peluang terdaftar.
Judul Solusi	Nama solusinya.
ID Solusi	Pengidentifikasi unik dari solusi.
ID Referensi Pembayar	Pengenal unik yang bukan ID akun. Ini terkait dengan akun tempat biaya ditagih. Ini membantu melacak penggunaan, pendapatan, dan langganan oleh pelanggan di semua laporan AWS Marketplace keuangan.
ID Alamat Pembayar	Pengenal unik yang mewakili alamat pelanggan .
Tanggal jatuh tempo pembayaran	Tanggal jatuh tempo pembayaran dalam formatYYYY-MM-DD .

Laporan pendapatan bulanan yang ditagih

Important

Tiga bagian dari laporan ini tidak akan tersedia pada 30 Agustus 2024. Anda dapat menggunakan alternatif ini untuk bagian 1-3:

- Dasbor [pendapatan yang ditagih](#)
- Kueri SQL [billed_revenue_report aktif](#). GitHub

Bagian 4: Kontrak dengan jadwal pembayaran yang fleksibel akan tetap tersedia setelah 30 Agustus 2024.

Laporan pendapatan tagihan bulanan memberi Anda informasi otoritatif tentang pendapatan yang ditagih setiap bulan untuk tujuan akuntansi dan pelaporan keuangan lainnya. Laporan ini menunjukkan jumlah total yang AWS ditagih kepada pelanggan untuk penggunaan produk Anda setiap jam, tahunan, atau bulanan. Laporan ini memiliki empat bagian: jumlah tagihan untuk penggunaan per jam dan biaya bulanan, langganan panjang variabel, penggunaan demonstrasi lapangan, dan pembayaran fleksibel.

Important

Jumlah dalam laporan ini hanya mencerminkan pendapatan yang kami tagihkan kepada pelanggan, bukan jumlah yang kami kumpulkan.

Laporan ini tersedia di Portal Manajemen AWS Marketplace bawah tab Laporan. Jika Anda terdaftar di layanan analisis AWS Marketplace perdagangan, Anda dapat menggunakan panggilan API untuk menarik bagian laporan ini. Untuk informasi selengkapnya, lihat [the section called “AWS Marketplace Layanan Analisis Perdagangan”](#).

Note

Untuk membatalkan email notifikasi untuk laporan ini, hubungi tim [Operasi AWS Marketplace Penjual](#).

Jadwal publikasi

Laporan ini diterbitkan setiap bulan pada hari ke-15 setiap bulan pukul 00:00 UTC.

Data Penagihan dan Pendapatan mencakup dari tanggal 15 hingga 14 bulan depan.

Langganan jangka panjang mencakup hari pertama setiap bulan pada pukul 00:00 UTC hingga hari terakhir bulan itu pada pukul 23:59 UTC bulan sebelumnya.

Misalnya, laporan yang diterbitkan pada 15 Mei mencakup mulai 1 April pukul 00:00 UTC hingga 30 April pukul 23:59 UTC.

Topik

- [Bagian 1: Data penagihan dan pendapatan](#)
- [Bagian 2: Langganan panjang variabel](#)
- [Bagian 3: penggunaan demonstrasi AWS lapangan](#)
- [Bagian 4: Kontrak dengan jadwal pembayaran yang fleksibel](#)

Bagian 1: Data penagihan dan pendapatan

Bagian ini mencantumkan data untuk penagihan penggunaan, pengembalian uang, biaya, dan pajak penjualan dan penggunaan AS yang dikumpulkan.

Nama kolom	Deskripsi
ID Referensi Pelanggan	<p>Pengenal unik yang bukan ID akun. Ini membantu melacak penggunaan, pendapatan, dan langganan oleh pelanggan.</p> <p>ID referensi pelanggan dijadwalkan untuk penghentian pada 31 Desember 2025. Gunakan Akun AWS ID pelanggan sebagai gantinya.</p>
Nomor AWS Rekening Pelanggan	Akun AWS ID tempat tagihan ditagih.

Nama kolom	Deskripsi
Negara	Kode negara dua karakter yang dikaitkan dengan akun tempat tagihan ditagih. Laporan ini menggunakan standar ISO 3166-1 alpha-2.
Negara Bagian	Alamat penagihan menyatakan yang terkait dengan akun tempat tagihan ditagih.
Kota	Kota alamat penagihan yang terkait dengan akun yang dikenakan biaya ditagih.
Kode Pos	Kode pos alamat penagihan yang dikaitkan dengan akun tempat tagihan ditagih.
Judul Produk	Judul produk.
Kode Produk	Pengidentifikasi unik untuk produk.
Jumlah Tagihan Pelanggan	Jumlah yang ditagihkan kepada pelanggan untuk penggunaan atau biaya bulanan produk.
AWS Biaya Listing	Jumlah AWS Marketplace biaya yang akan dikurangkan dari jumlah yang ditagih.
Jumlah Pengembalian Uang	Jumlah total biaya berlangganan dikembalikan kepada pelanggan jika ada pengembalian uang yang diproses selama periode cakupan data.
AWS Pengembalian Biaya	Bagian dari AWS Marketplace biaya dikembalikan jika ada pengembalian uang yang diproses selama periode cakupan data.
Biaya	Biaya barang ke pengecer: misalnya, apa yang reseller membayar Anda ketika mereka menjual produk Anda.

Nama kolom	Deskripsi
Jumlah Pendapatan Mitra	Jumlah total yang ditagih untuk transaksi, setelah dikurangi AWS Marketplace biaya, pengembalian uang, dan pajak penjualan dan penggunaan AS.
Mata Uang	Mata uang transaksi. Misalnya, jika transaksi dalam dolar AS, entri adalah USD.
ID Referensi Transaksi	Pengidentifikasi unik untuk transaksi yang membantu Anda menghubungkan transaksi di seluruh AWS Marketplace laporan.
Jumlah Tagihan Pelanggan Pajak Penjualan AS	Jumlah total pajak penjualan dan penggunaan AS yang ditagih untuk transaksi ini atas nama Penjual. (Artinya, catatan terkait dalam laporan Penjualan dan Pajak AS menunjukkan “pihak yang bertanggung jawab pajak” == “PENJUAL” .)
Jumlah Pengembalian Pajak Penjualan AS	Jumlah total pajak penjualan dan penggunaan AS yang dikembalikan untuk transaksi ini jika pengembalian dana diproses, ketika pajak tersebut dikumpulkan atas nama Penjual. (Artinya, catatan terkait dalam laporan Penjualan dan Pajak AS menunjukkan “pihak yang bertanggung jawab pajak” == “PENJUAL” .)
ID Penawaran	Pengidentifikasi untuk penawaran yang ditandatangani pembeli.
Menawarkan Visibilitas	Apakah penawaran itu adalah penawaran kontrak publik, swasta, atau perusahaan.

Nama kolom	Deskripsi
Domain Email Pelanggan	Domain email yang dikaitkan dengan akun tempat tagihan ditagih. Misalnya, jika alamat emailnya <code>liu-jie@example.com</code> , entri adalah <code>example.com</code> .
Nama Peluang	Pengidentifikasi unik untuk peluang terdaftar.
Deskripsi Peluang	Metadata untuk peluang terdaftar.
Judul Solusi	Nama solusinya.
ID Solusi	Pengidentifikasi unik untuk solusinya.
ID Referensi Pembayar	Pengenal unik yang bukan ID akun. Ini terkait dengan akun tempat biaya ditagih. Ini membantu melacak penggunaan, pendapatan, dan langganan oleh pelanggan di semua laporan AWS Marketplace keuangan.
ID Alamat Pembayar	Pengidentifikasi unik yang mewakili alamat pelanggan.

Bagian 2: Langganan panjang variabel

Bagian ini mencantumkan data untuk biaya berbasis biaya.

Nama kolom	Deskripsi
ID Referensi Pelanggan	<p>Pengenal unik yang bukan ID akun. Ini membantu melacak penggunaan, pendapatan, dan langganan oleh pelanggan.</p> <p>ID referensi pelanggan dijadwalkan untuk penghentian pada 31 Desember 2025. Gunakan Akun AWS ID pelanggan sebagai gantinya.</p>

Nama kolom	Deskripsi
Nomor AWS Rekening Pelanggan	Akun AWS ID tempat tagihan ditagih.
Negara	Kode negara dua karakter yang dikaitkan dengan akun tempat tagihan ditagih. Laporan ini menggunakan standar ISO 3166-1 alpha-2.
Negara Bagian	Alamat penagihan menyatakan yang terkait dengan akun tempat tagihan ditagih.
Kota	Kota alamat penagihan yang terkait dengan akun yang dikenakan biaya ditagih.
Kode Pos	Kode pos alamat penagihan yang dikaitkan dengan akun tempat tagihan ditagih.
Judul Produk	Judul produk.
Kode Produk	Pengidentifikasi unik untuk produk.
Kuantitas Berlangganan	Jumlah total lisensi yang ditentukan sebagai bagian dari pembelian berlangganan panjang variabel.
Tanggal Mulai langganan	Tanggal mulai pembelian berlangganan dengan panjang variabel.
Tanggal Berakhir Berlangganan	Tanggal akhir pembelian berlangganan dengan panjang variabel.
Jenis Instans Berlangganan	Jenis instans yang terkait dengan pembelian langganan panjang variabel.
Jumlah Tagihan Pelanggan	Jumlah yang ditagih untuk penggunaan, biaya bulanan, atau keduanya.
AWS Biaya Listing	Jumlah AWS Marketplace biaya yang dikurangkan dari jumlah yang ditagih.

Nama kolom	Deskripsi
Jumlah Pengembalian Uang	Jumlah total yang dikembalikan kepada pelanggan jika ada pengembalian uang yang diproses selama periode cakupan data.
AWS Pengembalian Biaya	Bagian dari AWS Marketplace biaya dikembalikan jika ada pengembalian uang yang diproses selama periode cakupan data.
Biaya	Biaya barang ke pengecer: misalnya, apa yang reseller membayar Anda ketika mereka menjual produk Anda.
Jumlah Pendapatan Mitra	Jumlah total yang ditagih untuk transaksi ini, setelah dikurangi AWS Marketplace biaya, pengembalian uang, dan pajak penjualan dan penggunaan AS.
Mata Uang	Mata uang transaksi. Misalnya, jika transaksi dalam dolar AS, entri adalah USD.
ID Referensi Transaksi	Pengidentifikasi unik untuk transaksi yang membantu Anda menghubungkan transaksi di seluruh AWS Marketplace laporan.
Jumlah Tagihan Pelanggan Pajak Penjualan AS	Jumlah total pajak penjualan dan penggunaan AS yang ditagih untuk transaksi ini atas nama Penjual. (Artinya, catatan terkait dalam laporan Penjualan dan Pajak AS menunjukkan “pihak yang bertanggung jawab pajak” == “PENJUAL” .)

Nama kolom	Deskripsi
Jumlah Pengembalian Pajak Penjualan AS	Jumlah total pajak penjualan dan penggunaan AS yang dikembalikan untuk transaksi ini jika pengembalian dana diproses, ketika pajak tersebut dikumpulkan atas nama Penjual. (Artinya, catatan terkait dalam laporan Penjualan dan Pajak AS menunjukkan “pihak yang bertanggung jawab pajak” == “PENJUAL” .)
Domain Email Pelanggan	Domain email yang dikaitkan dengan akun tempat tagihan ditagih. Misalnya, jika alamat emailnya <code>liu-jie@example.com</code> , entri adalah <code>example.com</code> .
ID Penawaran	Pengidentifikasi untuk penawaran yang ditandatangani pembeli.
Menawarkan Visibilitas	Apakah penawaran itu adalah penawaran kontrak publik, swasta, atau perusahaan.
Tanggal Mulai Kontrak	Tanggal mulai untuk kontrak AWS Marketplace SaaS.
Tanggal Akhir Kontrak	Tanggal akhir untuk kontrak AWS Marketplace SaaS.
Nama Peluang	Pengidentifikasi unik untuk peluang terdaftar.
Deskripsi Peluang	Metadata untuk peluang terdaftar.
Judul Solusi	Nama solusinya.
ID Solusi	Pengidentifikasi unik untuk solusinya.

Nama kolom	Deskripsi
ID Referensi Pembayar	Pengenalan unik yang bukan ID akun. Ini terkait dengan akun tempat biaya ditagih. Ini membantu melacak penggunaan, pendapatan, dan langganan oleh pelanggan di semua laporan AWS Marketplace keuangan.
ID Alamat Pembayar	Pengidentifikasi unik yang mewakili alamat pelanggan.

Bagian 3: penggunaan demonstrasi AWS lapangan

Bagian ini mencantumkan data untuk [penggunaan demonstrasi AWS lapangan](#) produk Anda. Anda dapat mengonfigurasi produk Anda untuk memungkinkan kami mendemonstrasikan produk Anda kepada calon pelanggan. Setiap penggunaan dari demonstrasi tercantum di sini.

Nama kolom	Deskripsi
Judul Produk	Judul produk.
Kode Produk	Pengidentifikasi unik untuk produk.
Tipe Instans	Jenis EC2 instans Amazon yang terkait dengan demonstrasi bidang.
Unit Penggunaan	Jumlah unit penggunaan yang terkait dengan produk.
Jenis Unit Penggunaan	Unit penggunaan yang terkait dengan jumlah unit penggunaan: misalnya, jam.

Bagian 4: Kontrak dengan jadwal pembayaran yang fleksibel

Bagian ini mencantumkan data untuk semua kontrak yang Anda buat dengan jadwal pembayaran yang fleksibel pada periode pelaporan sebelumnya.

Nama kolom	Deskripsi
Nomor AWS Rekening Pelanggan	Akun AWS ID tempat tagihan ditagih.
Negara Pelanggan	Kode negara dua karakter yang dikaitkan dengan akun pembayar tempat tagihan ditagih. Laporan ini menggunakan standar ISO 3166-1 alpha-2.
Negara Pelanggan	Alamat penagihan menyatakan yang terkait dengan akun pembayar tempat tagihan ditagih.
Kota Pelanggan	Kota alamat penagihan yang terkait dengan akun pembayar yang dikenakan biaya ditagih.
Kode Pos Pelanggan	Kode pos alamat penagihan yang dikaitkan dengan akun pembayar tempat tagihan ditagih.
Domain Email Pelanggan	Domain email yang terkait dengan akun pembayar tempat tagihan ditagih. Misalnya, jika alamat emailnya <code>liu-jie@example.com</code> , entri adalah <code>example.com</code> .
ID Referensi Pengguna	Akun akun pembayar tempat tagihan ditagih.
Nomor AWS Akun Pengguna	ID akun yang berlangganan produk.
ID Produk	Pengidentifikasi unik untuk produk.
Judul Produk	Judul produk.
Jenis Produk	Jenis produk.
AWS Marketplace ID Penawaran	Pengidentifikasi untuk penawaran yang ditandatangani pembeli.
Tanggal Membuat Kontrak	Tanggal pembuatan kontrak, yang merupakan tanggal akun berlangganan penawaran.
Tanggal Kedaluwarsa Kontrak	Tanggal ketika kontrak berakhir.

Nama kolom	Deskripsi
Total Nilai Kontrak (USD)	Nilai total kontrak dalam USD.
# Pembayaran	Jumlah pembayaran yang dijadwalkan untuk kontrak.
Tanggal Faktur	Tanggal faktur dibuat.
Jumlah Faktur (USD)	Jumlah yang ditagih pada faktur dalam USD.
ID Referensi Pembayar	Pengenal unik yang bukan ID akun. Ini terkait dengan akun tempat biaya ditagih. Ini membantu melacak penggunaan, pendapatan, dan langganan oleh pelanggan di semua laporan AWS Marketplace keuangan.

Laporan kompensasi penjualan

Important

Laporan ini tidak akan tersedia pada 30 Agustus 2024. Anda dapat menggunakan alternatif ini:

- Dasbor [pendapatan yang ditagih](#)
- Kueri [billed_revenue_report aktif](#)SQL. GitHub

Laporan ini mencantumkan pendapatan tagihan bulanan dengan informasi pelanggan tambahan yang tidak sesuai standar [the section called “Laporan pendapatan bulanan yang ditagih”](#). Laporan ini mencantumkan jumlah total yang AWS ditagih kepada pelanggan untuk penggunaan produk Anda setiap jam, tahunan, atau bulanan.

Important

Jumlah dalam laporan ini hanya mencerminkan pendapatan yang ditagihkan kepada pelanggan, bukan jumlah yang dikumpulkan dari pelanggan.

Laporan kompensasi penjualan dan informasi yang dibagikan dengan Anda sebagai bagian dari program ini merupakan Informasi Rahasia Amazon berdasarkan perjanjian kerahasiaan kami dengan Anda atau, jika tidak ada perjanjian tersebut, Syarat dan Ketentuan untuk penjual. AWS Marketplace Anda dapat menggunakan informasi ini hanya untuk mengkompensasi perwakilan penjualan Anda dengan memetakan AWS Marketplace pendapatan kepada perwakilan berdasarkan nama perusahaan, geografi, dan AWS ID akun. Anda dapat membagikan informasi ini dengan karyawan yang perlu mengetahuinya untuk memahami sumber komisi yang dibayarkan kepada mereka. Penggunaan dan pembagian informasi tersebut oleh Anda harus mematuhi kewajiban dalam perjanjian kerahasiaan kami dengan Anda dan syarat dan ketentuan untuk AWS Marketplace penjual, termasuk, namun tidak terbatas pada, Bagian 3.8 dari Syarat dan Ketentuan untuk penjual. AWS Marketplace

Untuk membuat laporan Anda sendiri yang mirip dengan yang satu ini AWS Marketplace [Umpan data pengiriman penjual di AWS Marketplace](#), Anda dapat menggunakan, termasuk [Laporan kompensasi penjualan](#) sebagai laporan dasar untuk menyesuaikan untuk memenuhi kebutuhan Anda.

Jadwal publikasi

Laporan ini diterbitkan setiap bulan, pada hari kelima belas setiap bulan pukul 00:00. UTC Laporan ini mencakup bulan kalender sebelumnya dari pukul 00:00 UTC hingga 23:59 UTC dari hari kalender terakhir bulan itu. Misalnya, laporan yang diterbitkan pada 15 Mei mencakup mulai 1 April pukul 00:00 UTC hingga 30 April pukul UTC 23:59.

Pendapatan yang ditagih

Bagian pendapatan yang ditagih dari laporan ini mencakup penggunaan dan biaya berbasis biaya dari bulan kalender sebelumnya. Berikut ini adalah nama kolom dan deskripsi.

Note

Dalam laporan ini, biaya listing adalah persentase dari hasil transaksi (kecuali untuk yang dijual kembali oleh reseller resmi produk penjualan kembali resmi) yang ditentukan sesuai dengan biaya listing berjenjang.

Nama kolom	Deskripsi
Nomor AWS Rekening Pelanggan	Akun tempat tagihan ditagih.

Nama kolom	Deskripsi
Negara	Kode negara dua karakter yang dikaitkan dengan akun tempat tagihan ditagih. Laporan ini menggunakan standar ISO alfa-2 3166-1.
Negara Bagian	Alamat penagihan menyatakan yang terkait dengan akun tempat tagihan ditagih.
Kota	Kota alamat penagihan yang terkait dengan akun tempat tagihan ditagih.
Kode Pos	Kode pos alamat penagihan yang dikaitkan dengan akun tempat tagihan ditagih.
Domain Email	Domain email yang dikaitkan dengan akun tempat tagihan ditagih. Misalnya, jika alamat emailnya <code>aliu-jie@example.com</code> , entri adalah <code>example.com</code> .
Kode Produk	Pengidentifikasi unik untuk produk.
Judul Produk	Judul produk.
Pendapatan Bruto	Jumlah yang ditagih untuk menggunakan produk atau biaya bulanan untuk menggunakan produk.
AWS Bagi Pendapatan	Jumlah AWS biaya yang dikurangkan dari jumlah yang ditagih pada waktu penyelesaian. Itu muncul di the section called "Laporan pencairan" .
Pengembalian Bruto	Jumlah total pengembalian uang untuk transaksi.
AWS Pengembalian Uang Berbagi	Bagian dari AWS biaya yang dikembalikan untuk transaksi.

Nama kolom	Deskripsi
Pendapatan Bersih	Jumlah bersih yang ditagih untuk transaksi ini, dikurangi AWS biaya, pengembalian uang, dan pajak penjualan dan penggunaan AS.
Mata Uang	Mata uang transaksi. Misalnya, jika transaksi dalam dolar AS, entri adalah USD.
Periode AR	Bulan dan tahun transaksi, dalam format YYYY-MM.
ID Referensi Transaksi	Pengidentifikasi unik yang mewakili transaksi, yang dapat Anda gunakan untuk menghubungkan transaksi di seluruh AWS Marketplace laporan.
Nama Peluang	Pengidentifikasi unik untuk peluang terdaftar.
Deskripsi Peluang	Setiap metadata dalam peluang terdaftar.
Judul Solusi	Nama solusinya.
ID Solusi	Pengidentifikasi unik dari solusi.
ID Referensi Pembayar	Pengenal unik yang bukan ID akun. Ini terkait dengan akun tempat biaya ditagih. Ini membantu melacak penggunaan, pendapatan, dan langganan oleh pelanggan di semua laporan AWS Marketplace keuangan.
ID Alamat Pembayar	Pengidentifikasi unik yang mewakili alamat pelanggan.

Laporan tambahan

AWS Marketplace mengirimkan laporan tambahan melalui akun Amazon S3 [Umpan data pengiriman penjual di AWS Marketplace](#) milik penjual yang terhubung ke ID Akun Penjual yang terkait dengan

daftar AWS Marketplace untuk penjual. AWS Marketplace Untuk informasi selengkapnya, lihat [Bucket Layanan Penyimpanan Sederhana Amazon tujuan](#).

Laporan tambahan diterbitkan setiap hari pada pukul 16:00 UTC jika ada pelanggan baru pada hari sebelumnya. Laporan ini mencakup hari sebelumnya dari 13:59 UTC hingga 16:01 UTC pada hari berikutnya.

Laporan detail perjanjian

Laporan detail perjanjian membantu Anda mendukung pelanggan yang menggunakan uji coba gratis kontrak perangkat lunak sebagai layanan (SaaS). Laporan tersebut mencakup rincian perjanjian seperti nama pelanggan, ID pelanggan, ID penawaran, mulai perjanjian, dan tanggal akhir perjanjian.

Anda hanya menerima laporan ini jika informasi yang relevan tersedia. Jika Anda tidak menerima laporan ini pada suatu kesempatan ketika Anda berpikir bahwa Anda harus, hubungi tim [Operasi AWS Marketplace Penjual](#).

Anda dapat mengakses laporan ini melalui bucket Amazon S3 yang terkait dengan ID Akun AWS Marketplace Penjual.

Tabel berikut mencantumkan nama kolom dan deskripsi untuk laporan detail perjanjian.

Data laporan uji coba gratis kontrak SaaS

Nama	Penjelasan
vendor_display_name	Nama vendor yang menjual produk.
vendor_aws_account_id	Identifikasi yang terkait dengan vendor yang menjual produk.
subscriber_aws_account_id	Identifikasi Akun AWS yang terkait dengan yang berlangganan produk.
customer_id	Pengidentifikasi unik untuk produk perangkat lunak.
product_title	Judul produk.
offer_id	Pengidentifikasi untuk penawaran yang ditandatangani pembeli.

Nama	Penjelasan
offer_visibility	Indikasi apakah penawaran tersebut merupakan penawaran kontrak publik, swasta, atau perusahaan.
reseller_name	Nama reseller mitra saluran.
reseller_aws_account_id	Pengenal unik untuk reseller mitra saluran.
agreement_id	Referensi umpan data perjanjian unik untuk perjanjian yang ditandatangani antara pengusul dan penerima untuk mulai menggunakan produk.
agreement_acceptance_date	Tanggal perjanjian diterima.
agreement_start_date	Tanggal mulai perjanjian.
agreement_end_date	Tanggal akhir perjanjian. Untuk metered/pay as go/subscriptions, ini diatur ke 1-JAN-9999.
is_free_trial_offer	Bendera yang menunjukkan apakah penawaran atau perjanjian adalah penawaran uji coba gratis.
is_upgraded_after_free_trial	Bendera yang menunjukkan apakah perjanjian ditingkatkan ke kontrak berbayar.
total_contract_value	Nilai total kontrak.

Dasbor penjual

AWS Marketplace menyediakan dasbor yang didukung oleh [Amazon QuickSight](#) dengan bagan, grafik, dan wawasan yang membantu Anda mengakses dan menganalisis data keuangan, penjualan, dan pemasaran. Dasbor penjual meliputi:

[the section called “Dasbor untuk operasi keuangan”](#)

- [the section called “Dasbor pendapatan yang ditagih”](#)— Memberikan informasi tentang pendapatan yang ditagih untuk tujuan akuntansi dan pelaporan keuangan lainnya.
- [the section called “Dasbor koleksi dan pencairan”](#)— Memberikan informasi tentang dana yang AWS dikumpulkan dan dicairkan ke rekening bank Anda sejak pencairan sebelumnya.
- [the section called “Dasbor perpajakan”](#)— Memberikan informasi tentang pajak untuk transaksi penjual.

[Dasbor untuk operasi penjualan](#)

- [the section called “Dasbor perjanjian dan pembaruan”](#)— Memberikan informasi tentang perjanjian dan pembaruan dalam waktu 24 jam setelah penandatanganan perjanjian. AWS Marketplace
- [the section called “Dasbor penggunaan”](#)— Menyediakan visualisasi dan data berbutir halus untuk pelanggan yang menggunakan SaaS dan produk berbasis penggunaan server.
- [Dasbor untuk pemasaran](#)— Menyediakan beberapa dasbor untuk membantu Anda melacak data pemasaran Anda.

[Dasbor untuk pemasaran](#)

- [Dasbor perjanjian pelanggan](#)— Memberikan data tentang perjanjian dan pelanggan yang berlangganan produk Anda.
- [Daftar dasbor kinerja](#)— Menyediakan data tentang lalu lintas dan perilaku pengguna dalam AWS Marketplace daftar Anda.
- [Dasbor kinerja pencarian](#)— Menyediakan data tentang kata kunci yang diterapkan pada AWS Marketplace daftar.

Dasbor tersedia untuk AWS Marketplace penjual yang memiliki izin yang sesuai.

Mengakses dasbor

Secara default, administrator AWS Marketplace sistem untuk akun penjual memiliki akses ke semua dasbor pada tab Wawasan di. Portal Manajemen AWS Marketplace Administrator sistem dapat membuat kebijakan AWS Identity and Access Management (IAM) untuk menyediakan akses dasbor tertentu ke pengguna lain di perusahaan penjual.

Note

Pada bulan September 2023, kami tidak akan lagi mendukung akses ke dasbor penjual yang diaktifkan oleh izin IAM lama. Perbarui izin IAM Anda menggunakan format Amazon Resource Name (ARN) baru dalam contoh kode di bawah ini.

Untuk informasi tentang membuat kebijakan, lihat [Membuat kebijakan IAM](#).

Kebijakan dasbor

Gunakan salah satu kebijakan berikut untuk menyediakan akses ke beberapa atau semua dasbor.

Contoh berikut menyediakan akses ke AWS Marketplace sumber daya saat ini dan masa depan, termasuk dasbor dan laporan. Ini menggunakan semua feed data saat ini dan masa depan:

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": [
      "aws-marketplace:GetSellerDashboard"
    ],
    "Resource": [
      "arn:aws:aws-marketplace::awsAccountID:AWSMarketplace/*",
    ]
  }]
}
```

Anda juga dapat memberikan akses ke satu atau beberapa dasbor dengan memasukkannya. ARNs Contoh berikut menunjukkan cara menyediakan akses ke semua dasbor:

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": [
      "aws-marketplace:GetSellerDashboard"
    ],
    "Resource": [
      "arn:aws:aws-marketplace::awsAccountID:AWSMarketplace/ReportingData/BillingEvent_V1/Dashboard/BilledRevenue_V1",
    ]
  }]
}
```

```
"arn:aws:aws-marketplace::awsAccountID:AWSMarketplace/ReportingData/  
BillingEvent_V1/Dashboard/CollectionsAndDisbursements_V1",  
"arn:aws:aws-marketplace::awsAccountID:AWSMarketplace/ReportingData/Agreement_V1/  
Dashboard/AgreementsAndRenewals_V1"  
"arn:aws:aws-marketplace::awsAccountID:AWSMarketplace/ReportingData/Usage_V1/  
Dashboard/Usage_V1"  
"arn:aws:aws-marketplace::awsAccountID:AWSMarketplace/ReportingData/TaxItem_V1/  
Dashboard/Tax_V1"  
"arn:aws:aws-marketplace::awsAccountID:AWSMarketplace/ReportingData/Marketing_V1/  
Dashboard/CustomerAgreements_V1"  
"arn:aws:aws-marketplace::awsAccountID:AWSMarketplace/ReportingData/Marketing_V1/  
Dashboard/ListingPerformance_V1"  
"arn:aws:aws-marketplace::awsAccountID:AWSMarketplace/ReportingData/Marketing_V1/  
Dashboard/SearchPerformance_V1"  
  ]  
}]  
}
```

Untuk menghapus akses ke dasbor, hapus dari Resource bagian kebijakan.

Note

Untuk informasi tentang membuat kebijakan AWS Identity and Access Management (IAM), lihat [Membuat kebijakan IAM](#) di AWS Identity and Access Management Panduan Pengguna.

Dasbor untuk operasi keuangan

AWS Marketplace menyediakan beberapa dasbor untuk membantu Anda melacak data keuangan Anda.

Topik

- [Dasbor pendapatan yang ditagih](#)
- [Dasbor koleksi dan pencairan](#)
- [Dasbor perpajakan](#)

Dasbor pendapatan yang ditagih

Dasbor pendapatan yang ditagih memberikan informasi tentang semua penjualan yang ditagih di AWS Marketplace Laporan ini diharapkan dapat menghemat waktu dan tersedia sekitar 45 hari lebih

awal dari laporan [pendapatan tagihan bulanan](#) lama. Laporan warisan disampaikan pada hari ke-15 setiap bulan, yang menunda visibilitas penagihan bulan sebelumnya.

Untuk membuka dasbor ini, masuk ke [Portal Manajemen AWS Marketplace](#), pilih Wawasan, Operasi keuangan, lalu pilih tab Pendapatan yang ditagih.

Topik

- [Segarkan frekuensi dasbor pendapatan yang ditagih](#)
- [Bagian 1: Kontrol](#)
- [Bagian 2: Filter](#)
- [Bagian 3: Metrik utama](#)
- [Bagian 4: Tren](#)
- [Bagian 5: Kerusakan](#)
- [Bagian 6: Data granular](#)

Segarkan frekuensi dasbor pendapatan yang ditagih

Dasbor diperbarui setiap hari pada tengah malam UTC. Jika faktur dibuat pada 10 Mei sebelum tengah malam UTC, pembaruan pada 11 Mei akan menampilkan faktur. Jika faktur dibuat pada 10 Mei setelah tengah malam UTC, maka pembaruan pada 12 Mei akan menampilkan faktur. Jika data faktur atau pencairan terbaru yang diterima dari sistem hulu tertunda, mungkin ada penundaan 1-2 hari agar data terbaru tercermin di dasbor.

Jika Anda tidak melihat faktur pelanggan yang diharapkan, gunakan prosedur berikut sebelum menghubungi kontak pengembangan AWS Marketplace bisnis Anda.

Untuk memecahkan masalah faktur pelanggan yang hilang untuk dasbor pendapatan yang ditagih

1. Konfirmasikan bahwa penawaran diterima oleh pelanggan menggunakan satu atau lebih alat berikut:
 - [Laporan pelanggan pelanggan harian](#)
 - [Pemberitahuan untuk AWS Marketplace acara](#)
 - [Portal Manajemen AWS Marketplace tab perjanjian](#)
2. (Untuk penawaran pribadi) Setelah Anda mengonfirmasi bahwa pelanggan menerima penawaran, tinjau jadwal dan jumlah faktur penawaran:

- Untuk penawaran pribadi, periksa tab Portal Manajemen AWS Marketplace Penawaran.
 - Untuk penawaran pribadi mitra saluran, periksa tab Portal Manajemen AWS Marketplace Mitra untuk melihat jadwal pembayaran khusus dalam peluang mitra.
3. Pertimbangkan apakah transaksi menggunakan [model penetapan harga penggunaan](#), di mana pelanggan ditagih pada hari kedua dan ketiga setiap bulan untuk bulan penggunaan sebelumnya.

Bagian 1: Kontrol

Bagian dasbor ini menyediakan filter untuk menyempurnakan data dasbor pendapatan yang ditagih. Misalnya, Anda dapat memilih filter pada bidang dari [notifikasi AWS Marketplace acara guna](#) mengonfirmasi penagihan ID akun pelanggan tertentu, nama perusahaan pelanggan, atau ID penawaran. Anda juga dapat menambahkan filter ke analisis, seperti rentang tanggal yang ingin Anda sertakan dalam visual. Filter yang dipilih dalam kontrol memperbarui data yang ditampilkan di bagian metrik, tren, gangguan, dan data granular.

Untuk informasi selengkapnya tentang pemfilteran, lihat [Memfilter data QuickSight di QuickSight](#) Panduan Pengguna.

Deskripsi kontrol

Nama kontrol	Deskripsi
ID Pelanggan Akun AWS	ID akun yang berlangganan produk.
Nama perusahaan pelanggan	Nama akun yang berlangganan produk.
Judul produk	Judul produk.
ID Penawaran	Pengidentifikasi untuk penawaran yang ditandatangani pembeli.
Menawarkan visibilitas	Apakah penawaran itu adalah penawaran kontrak publik, swasta, atau perusahaan.
ID Perjanjian	Referensi umpan data perjanjian unik untuk perjanjian yang ditandatangani antara pengusul

Nama kontrol	Deskripsi
	dan penerima untuk mulai menggunakan produk.
AWS penjual catatan	<p>Pengenal entitas bisnis yang memfasilitasi transaksi. Nilai yang mungkin termasuk:</p> <ul style="list-style-type: none"> • AWS_INC: Pengidentifikasi untuk AWS, Inc. (berbasis di Amerika Serikat). • AWS_EUROPE: Pengidentifikasi untuk AWS EMEA SARL (berbasis di Luksemburg). • AWS_AUSTRALIA: Pengidentifikasi untuk AWS Australia Pty Ltd • AWS_JAPAN: Pengidentifikasi untuk AWS Jepang G.K. • AWS_KOREA: Pengidentifikasi untuk AWS Korea.
ID Pembayar Akun AWS	ID akun tempat tagihan ditagih.
Nama perusahaan pembayar	Nama bisnis akun tempat tagihan ditagih.
Perusahaan pengecer	Nama bisnis akun pengecer yang berwenang untuk menjual produk produsen perangkat lunak.
ID Pengecer Akun AWS	ID akun yang membeli produk atau layanan secara grosir dari ISV untuk dijual kembali ke pelanggan.
ID otorisasi penjualan kembali	Pengidentifikasi unik untuk peluang terdaftar.
Nama otorisasi penjualan kembali	Nama unik untuk peluang terdaftar.
Negara pelanggan	Kode negara dua karakter yang terkait dengan akun berlangganan produk.

Nama kontrol	Deskripsi
Negara atau wilayah pelanggan	Status alamat penagihan atau wilayah yang terkait dengan akun berlangganan produk.
ID referensi transaksi	Pengenal unik untuk transaksi yang membantu Anda mengkorelasikan transaksi di seluruh laporan AWS Marketplace lama.

Bagian 2: Filter

Bagian dasbor ini menyediakan filter untuk menyempurnakan catatan berdasarkan mata uang penawaran dan rentang tanggal faktur. Anda dapat menggunakan rentang tanggal faktur untuk memilih apakah tanggal penagihan sebelum atau setelah tanggal yang ditentukan atau dalam rentang tanggal. Filter yang dipilih memperbarui data yang ditampilkan di bagian metrik, tren, perincian, dan data granular. Rentang tanggal faktur default diatur untuk menarik data dari enam bulan terakhir.

Bagian 3: Metrik utama

Bagian dasbor ini menampilkan indikator kinerja utama (KPI) untuk memvisualisasikan perbandingan angka pendapatan utama yang ditagih menggunakan filter yang dipilih. KPI ditampilkan untuk pendapatan kotor, pengembalian dana kotor, biaya daftar, biaya grosir (jika ada), dan pendapatan bersih penjual untuk rentang tanggal faktur dan mata uang yang ditentukan. Anda dapat memperbarui kriteria filter untuk melihat metrik mata uang dan tanggal yang berbeda.

Bagian 4: Tren

Bagian dasbor ini menyediakan tren pendapatan yang ditagih untuk rentang tanggal yang ditentukan. Anda dapat melihat tren berdasarkan agregasi tanggal yang ditentukan—seperti hari,, month-over-month quarter-over-quarter, atau year-over-year —untuk mendapatkan wawasan tentang pendapatan yang ditagih. Informasi tren pendapatan yang ditagih tersedia pada tingkat agregat atau berdasarkan jenis visibilitas penawaran:

- Tren penagihan - Memberikan gambaran pendapatan kotor, pendapatan bersih penjual, biaya grosir (jika ada), dan pengembalian uang dari waktu ke waktu untuk rentang tanggal yang dipilih dari filter agregasi tanggal.

- Tren penagihan berdasarkan visibilitas penawaran — Memberikan gambaran jumlah penawaran dan pendapatan kotor berdasarkan jenis visibilitas penawaran dari waktu ke waktu di seluruh [penawaran pribadi](#), [penawaran publik](#) (atau layanan mandiri), dan program perusahaan.

Bagian 5: Kerusakan

Bagian dasbor ini memberi Anda metrik utama tentang bisnis Anda di seluruh pelanggan, penjual catatan, geografi pelanggan, dan judul produk. Anda dapat memfilter berdasarkan pendapatan kotor, jumlah pembayar, jumlah pelanggan, pengembalian dana kotor, biaya daftar, pendapatan bersih penjual, dan biaya grosir.

Bagian 6: Data granular

Bagian dasbor ini menunjukkan semua penjualan yang ditagih, termasuk jumlah total yang AWS ditagih kepada pelanggan untuk penggunaan produk Anda setiap jam, tahunan, atau bulanan. AWS tagihan pelanggan menggunakan tiga frekuensi berikut:

- Tanggal penerimaan langganan (tagihan di muka)
- Jadwal pembayaran khusus (penawaran pribadi yang dibangun oleh penjadwal pembayaran fleksibel)
- Penggunaan terukur pada hari kedua dan ketiga bulan itu untuk penggunaan bulan sebelumnya

Note

Faktur yang dibuat sebelum 1 April 2021 mungkin tidak memiliki ID perjanjian terkait, ID penawaran, Akun AWS ID pelanggan, atau nama perusahaan pelanggan.

Biaya pelanggan ditampilkan di bagian data granular ini 24 jam setelah pelanggan ditagih. Misalnya, jika pelanggan ditagih pada 3 November, maka dasbor melaporkan faktur ini pada 4 November. Untuk informasi selengkapnya tentang cara mengekspor dan mengunduh data dari QuickSight tabel, lihat [Mengekspor data dari visual](#) di QuickSight Panduan Pengguna.

Deskripsi data granular

Kolom	Deskripsi
Tanggal faktur	Tanggal pelanggan ditagih untuk berlangganan produk.
Tanggal jatuh tempo pembayaran	Tanggal jatuh tempo pembayaran dalam format YYYY-MM-DD.
Ketentuan pembayaran	Ketentuan pembayaran AWS faktur pelanggan.
ID Faktur	AWS ID yang ditetapkan ke AWS faktur tempat tagihan ditagih.
ID faktur biaya daftar	Ketika AWS Marketplace langganan ditransaksikan melalui AWS EMEA SARL, badan hukum Jepang atau Australia (penjual catatan), operator untuk penjualan diharuskan membebankan pajak pertambahan nilai (PPN) kepada penjual atas biaya daftar penjual dan diberikan faktur yang sesuai dengan pajak. Untuk transaksi yang berlaku, ID faktur untuk PPN yang dinilai pada biaya listing berbeda dari ID faktur langganan perangkat lunak atau produk. Transaksi dari AWS, Inc. memiliki nilai "MISSING_LISTING_FEE_INVOICE_ID" karena ID faktur biaya listing tidak berlaku.
ID faktur grosir	AWS ID yang ditetapkan ke faktur yang tidak dapat dibayar yang mewakili penjualan antara ISV dan Mitra Saluran dalam penawaran Pribadi Mitra Saluran (CPPO). Untuk penawaran umum dan penawaran Marketplace Private (MPOs) bidang ini akan menjadi 'Tidak berlaku'.
Nama perusahaan pelanggan	Nama akun yang berlangganan produk.

Kolom	Deskripsi
ID Pelanggan Akun AWS	ID akun yang berlangganan produk.
Domain email pelanggan	Domain email yang terkait dengan akun yang berlangganan produk. Misalnya, jika alamat emailnya adalah liu-jie@example.com, entri adalah example.com.
Kota pelanggan	Kota alamat penagihan terkait dengan akun yang berlangganan produk.
Negara atau wilayah pelanggan	Status alamat penagihan yang terkait dengan akun yang berlangganan produk.
Negara pelanggan	Negara alamat penagihan yang terkait dengan akun yang berlangganan produk.
Kode pos pelanggan	Kode pos alamat penagihan terkait dengan akun yang berlangganan produk.
Judul produk	Judul produk.
Nama penawaran	Nama penawaran yang ditentukan penjual.
ID Penawaran	Pengidentifikasi untuk penawaran yang ditandatangani pembeli.
Menawarkan visibilitas	Apakah penawaran itu adalah penawaran kontrak publik, swasta, atau perusahaan.
ID Perjanjian	Referensi umpan data perjanjian unik untuk perjanjian yang ditandatangani antara pengusul dan penerima untuk mulai menggunakan produk.

Kolom	Deskripsi
Tanggal mulai perjanjian	Tanggal langganan produk pelanggan dimulai, diformat sebagai MM-DD-YYYY. Tanggal ini bisa berbeda dari tanggal penerimaan jika ini adalah perjanjian bertanggal masa depan.
Tanggal penerimaan perjanjian	Tanggal ketika pelanggan berlangganan produk, diformat sebagai MM-DD-YYYY
Tanggal akhir perjanjian	Tanggal ketika kontrak berakhir, diformat sebagai MM-DD-YYYY Untuk pay-as-you-go meter/langganan, tanggal ini diatur ke JAN-1-9999.
Tanggal akhir periode penggunaan	Tanggal akhir periode penggunaan produk.
Tanggal mulai periode penggunaan	Tanggal mulai periode penggunaan produk.
Status pencairan	Status yang terkait dengan faktur untuk mengonfirmasi bahwa AWS tidak ada dana, sebagian atau seluruh dana ke rekening bank Anda. Status yang mungkin adalah: Disbursed, Sebagian dicairkan, Gagal, Tidak dicairkan.
Tanggal pencairan	Tanggal AWS dimulai pencairan ke bank penjual.
Pencairan ID jejak bank	Untuk pencairan, ID jejak diberikan oleh bank. ID jejak bank pencairan dapat digunakan untuk menghubungkan pemberitahuan setoran yang disediakan bank penjual dan laporan ke faktur dalam laporan. AWS Marketplace
Pendapatan kotor	Jumlah yang ditagihkan kepada pelanggan untuk penggunaan atau biaya bulanan produk.

Kolom	Deskripsi
Pengembalian dana kotor	Jumlah total biaya berlangganan dikembalikan kepada pelanggan jika ada pengembalian uang yang diproses selama periode cakupan data.
Biaya listing	Jumlah AWS Marketplace biaya yang akan dikurangkan dari jumlah yang ditagih.
Pengembalian biaya listing	Bagian dari AWS Marketplace biaya dikembalikan jika ada pengembalian uang yang diproses selama periode cakupan data.
Persentase biaya daftar	Persentase AWS Marketplace biaya yang akan dikurangkan dari jumlah yang ditagih.
Pembagian pajak penjual	Jumlah total pajak penjualan dan penggunaan AS yang ditagih untuk transaksi ini.
Pengembalian pajak saham penjual	Jumlah total pajak penjualan dan penggunaan AS dikembalikan untuk transaksi ini jika pengembalian dana diproses.
AWS pembagian pajak	Jumlah total pajak penjualan dan penggunaan AS yang ditagih untuk transaksi ini atas nama penjual.
AWS pengembalian pajak saham	Jumlah total pajak penjualan dan penggunaan AS dikembalikan untuk transaksi ini jika pengembalian dana diproses, ketika pajak tersebut dikumpulkan atas nama penjual.
Biaya grosir	Hanya untuk penawaran pribadi mitra saluran. Biaya barang ke pengecer. Misalnya, apa yang dibayar pengecer kepada produsen ketika mereka menjual produk pabrikan. Biaya grosir adalah harga daftar dikalikan dengan persentase diskon.

Kolom	Deskripsi
Pengembalian biaya grosir	Hanya untuk penawaran pribadi mitra saluran. Biaya barang yang dikembalikan dari pengecer.
Pajak penjual grosir sahre	Pajak atas penjualan antara ISV dan Mitra Saluran dalam penawaran Pribadi Mitra Saluran (CPPO) di mana penjual bertanggung jawab pajak. Untuk penawaran umum dan penawaran Marketplace Private (MPPOs) bidang ini akan menjadi 'Tidak berlaku'.
Pengembalian saham pajak penjual grosir	Pengembalian pajak atas penjualan antara ISV dan Mitra Saluran dalam penawaran Pribadi Mitra Saluran (CPPO) di mana penjual bertanggung jawab. Untuk penawaran umum dan penawaran Marketplace Private (MPPOs) bidang ini akan menjadi 'Tidak berlaku'.
Grosir bagian pajak penjual lainnya	Pajak atas penjualan antara ISV dan Mitra Saluran dalam penawaran Pribadi Mitra Saluran (CPPO) di mana penjual bertanggung jawab pajak. Bidang ini diisi ketika penjual lain yang terlibat dalam transaksi bertanggung jawab. Untuk penawaran umum dan penawaran Marketplace Private (MPPOs) bidang ini akan menjadi 'Tidak berlaku'.
Pengembalian saham pajak penjual grosir lainnya	Pengembalian pajak atas penjualan antara ISV dan Mitra Saluran dalam penawaran Pribadi Mitra Saluran (CPPO) di mana penjual bertanggung jawab. Bidang ini diisi ketika penjual lain yang terlibat dalam transaksi bertanggung jawab. Untuk penawaran umum dan penawaran Marketplace Private (MPPOs) bidang ini akan menjadi 'Tidak berlaku'.

Kolom	Deskripsi
Pembagian AWS pajak grosir	Pajak atas penjualan antara ISV dan Mitra Saluran dalam penawaran Pribadi Mitra Saluran (CPPO) yang AWS dikenakan pajak. Untuk penawaran umum dan penawaran Marketplace Private (MPPOs) bidang ini akan menjadi 'Tidak berlaku'.
Pengembalian AWS pajak saham grosir	Pengembalian pajak atas penjualan antara ISV dan Mitra Saluran dalam penawaran Pribadi Mitra Saluran (CPPO) yang AWS dikenakan pajak. Untuk penawaran umum dan penawaran Marketplace Private (MPPOs) bidang ini akan menjadi 'Tidak berlaku'.
Pendapatan bersih penjual	Jumlah total yang ditagih untuk transaksi, setelah dikurangi AWS Marketplace biaya, pengembalian uang, dan pajak penjualan dan penggunaan AS.
Mata Uang	Mata uang transaksi. Misalnya, jika transaksi dalam dolar AS, entri adalah USD.
ID referensi transaksi	Pengidentifikasi unik yang mewakili transaksi , yang dapat Anda gunakan untuk menghubungkan transaksi di seluruh AWS Marketplace laporan.

Kolom	Deskripsi
AWS penjual catatan	<p>Pengenal entitas bisnis yang memfasilitasi transaksi. Kemungkinan nilainya adalah sebagai berikut:</p> <ul style="list-style-type: none">• AWS_INC: Pengenal untuk AWS, Inc. (berbasis di Amerika Serikat)• AWS_EUROPE: Pengidentifikasi untuk AWS EMEA SARL (berbasis di Luksemburg)• AWS AUSTRALIA: Pengidentifikasi untuk AWS Australia Pty Ltd• AWS_JAPAN: Pengidentifikasi untuk AWS Jepang G.K.• AWS_KOREA: Pengidentifikasi untuk Korea AWS
ID otorisasi penjualan kembali	Pengidentifikasi unik untuk peluang terdaftar.
Nama otorisasi penjualan kembali	Nama unik untuk peluang terdaftar.
Deskripsi otorisasi penjualan kembali	Deskripsi yang ditentukan ISV untuk peluang terdaftar.
Dijual kembali nama perusahaan	Nama akun yang membeli produk atau layanan dengan biaya grosir dari ISV untuk dijual kembali ke pelanggan.
ID Pembayar Akun AWS	ID akun yang membeli produk atau layanan dengan biaya grosir dari ISV untuk dijual kembali ke pelanggan.
Domain email pembayar	Domain email yang dikaitkan dengan akun tempat tagihan ditagih. Misalnya, jika alamat emailnya adalah liu-jie@example.com, entri adalah example.com.

Kolom	Deskripsi
Kota pembayar	Kota alamat penagihan yang terkait dengan akun yang dikenakan biaya ditagih.
Negara bagian atau wilayah pembayar	Alamat penagihan menyatakan terkait dengan akun tempat tagihan ditagih.
Negara pembayar	Kode negara dua karakter yang terkait dengan akun tempat tagihan ditagih.
Kode pos pembayar	Kode pos alamat penagihan yang terkait dengan akun tempat tagihan ditagih.
ID akun ISV	Pengidentifikasi pemilik produk atau layanan.
Nama perusahaan ISV	Nama bisnis pemilik produk atau layanan.
ID Produk	Pengidentifikasi unik yang ramah untuk produk perangkat lunak.
Pendapatan bersih yang dicairkan	Jumlah total untuk transaksi yang dicairkan kepada penjual. Jika jumlahnya tidak sama dengan "pendapatan bersih penjual", maka ini adalah pembayaran sebagian.
Pendapatan bersih yang belum dicairkan	Jumlah total transaksi yang tidak dicairkan kepada penjual. Jika jumlahnya selain nol, sisa saldo terutang oleh pelanggan.

Dasbor koleksi dan pencairan

Dasbor pengumpulan dan pencairan menyediakan data tentang dana yang dicairkan ke rekening bank Anda. Ini juga menyediakan daftar semua faktur di mana pencairan telah dikirim untuk pembayaran sebagian, pembayaran penuh, atau di mana pencairan gagal telah terjadi. Daftar ini tidak termasuk faktur di mana pencairan belum terjadi. Untuk meninjau faktur yang tidak dicairkan, gunakan dasbor Pendapatan yang Ditagih.

Pencairan termasuk pembayaran pelanggan, pengembalian uang yang diselesaikan untuk berlangganan produk Anda, dan beberapa pajak yang dikumpulkan atau dikembalikan kepada pelanggan. Pengembalian uang di dasbor muncul sebagai jumlah negatif karena uang dikembalikan ke pelanggan Anda setelah Anda mengotorisasi pengembalian dana.

Dasbor pengumpulan dan pencairan menyediakan akses lebih cepat ke pencairan pelanggan. Harapkan untuk menghemat sekitar empat hari dibandingkan dengan [laporan pencairan](#) warisan, yang dibuat lima hari setelah pencairan dikirim.

Note

Pelanggan memiliki persyaratan pembayaran yang berbeda dengan AWS, sehingga beberapa dana dalam kategori usia yang tidak ditagih mungkin tidak jatuh tempo dari pelanggan.

Dasbor pengumpulan dan pencairan memberikan informasi untuk proses operasional dan keuangan. Dasbor menyegarkan setiap hari. Untuk informasi selengkapnya, lihat topik berikut.

Untuk membuka dasbor, masuk ke [Portal Manajemen AWS Marketplace](#), pilih Wawasan, Operasi keuangan, lalu pilih tab Koleksi dan pencairan.

Topik

- [Segarkan frekuensi dasbor pengumpulan dan pencairan](#)
- [Bagian 1: Kontrol](#)
- [Bagian 2: Filter](#)
- [Bagian 3: Metrik utama](#)
- [Bagian 4: Tren](#)
- [Bagian 5: Kerusakan](#)
- [Bagian 6: Data granular](#)

Segarkan frekuensi dasbor pengumpulan dan pencairan

Dasbor koleksi dan pencairan diperbarui hanya pada hari kerja Amerika Utara. Anda dapat mengharapkan untuk melihat faktur yang dicairkan dalam satu hari setelah menerima setoran ke bank Anda.

Bagian 1: Kontrol

Bagian dasbor ini menyediakan filter untuk menyempurnakan data dasbor Anda. Misalnya, Anda dapat memilih filter pada bidang dari [notifikasi AWS Marketplace acara untuk](#) mengonfirmasi pencairan untuk ID akun pelanggan tertentu, nama perusahaan pelanggan, atau ID penawaran. Anda juga dapat memfilter berdasarkan status pencairan untuk memahami semua faktur yang dibayarkan kepada Anda atau faktur terbuka dan belum dibayar. Anda dapat menambahkan filter ke analisis, seperti rentang tanggal yang ingin Anda sertakan dalam visual. Filter yang dipilih dalam kontrol memperbarui data yang ditampilkan di bagian metrik, tren, perincian, dan data granular.

Untuk informasi selengkapnya tentang pemfilteran, lihat [Memfilter data QuickSight di QuickSight Panduan Pengguna](#).

Deskripsi kontrol

Nama kontrol	Deskripsi
ID Pelanggan Akun AWS	ID akun yang berlangganan produk.
Nama perusahaan pelanggan	Nama akun yang berlangganan produk.
Judul produk	Judul produk.
ID Penawaran	Pengidentifikasi untuk penawaran yang ditandatangani pembeli.
Menawarkan visibilitas	Apakah penawaran itu adalah penawaran kontrak publik, swasta, atau perusahaan.
ID Perjanjian	Referensi umpan data perjanjian unik untuk perjanjian yang ditandatangani antara pengusul dan penerima untuk mulai menggunakan produk.
AWS penjual catatan	<p>Pengidentifikasi entitas bisnis yang memfasilitasi transaksi. Nilai yang mungkin termasuk:</p> <ul style="list-style-type: none"> AWS_INC: Pengidentifikasi untuk AWS, Inc. (berbasis di Amerika Serikat)

Nama kontrol	Deskripsi
	<ul style="list-style-type: none"> • AWS_EUROPE: Pengidentifikasi untuk AWS EMEA SARL (berbasis di Luksemburg) • AWS_AUSTRALIA: Pengidentifikasi untuk AWS Australia Pty Ltd (Australia)AWS • AWS_JAPAN: Pengenal untuk AWS Jepang G.K. • AWS_KOREA: Pengidentifikasi untuk Korea AWS
Status pencairan	Status yang terkait dengan faktur untuk mengonfirmasi bahwa AWS tidak ada dana, sebagian atau seluruh dana ke rekening bank Anda. Status yang mungkin adalah: Disbursed, Sebagian dicairkan, Gagal, Tidak dicairkan.
ID Pembayar Akun AWS	ID akun tempat tagihan ditagih.
Nama perusahaan pembayar	Nama bisnis akun tempat tagihan ditagih.
Perusahaan pengecer	Nama bisnis akun pengecer yang berwenang untuk menjual produk produsen perangkat lunak.
ID Pengecer Akun AWS	ID akun yang membeli produk atau layanan secara grosir dari ISV untuk dijual kembali ke pelanggan.
ID otorisasi penjualan kembali	Pengidentifikasi unik untuk peluang terdaftar.
Nama otorisasi penjualan kembali	Nama unik untuk peluang terdaftar.
Negara pelanggan	Kode negara dua karakter yang terkait dengan akun berlangganan produk.
Negara atau wilayah pelanggan	Status alamat penagihan atau wilayah yang terkait dengan akun berlangganan produk.

Nama kontrol	Deskripsi
ID referensi transaksi	Pengidentifikasi unik untuk transaksi yang membantu Anda menghubungkan transaksi di seluruh laporan AWS Marketplace lama.
Pencairan ID jejak bank	Untuk pencairan, ID jejak diberikan oleh bank. ID jejak bank dapat mengkorelasikan pemberitahuan setoran yang disediakan bank penjual dan laporan ke faktur dalam laporan. AWS Marketplace

Bagian 2: Filter

Bagian dasbor ini menyediakan filter untuk menyempurnakan catatan berdasarkan mata uang penawaran dan dua dimensi tanggal yang berbeda — apakah nilai bidang tanggal sebelum atau setelah tanggal tertentu atau dalam rentang tanggal. Dimensi tanggal adalah tanggal jatuh tempo pembayaran atau tanggal pencairan terakhir. Tanggal pencairan membatasi hasil untuk faktur yang dicairkan dalam rentang tanggal yang ditentukan. Tanggal jatuh tempo pembayaran termasuk faktur dengan tanggal jatuh tempo dalam kisaran yang ditentukan, terlepas dari tanggal pencairan. Filter kategori tanggal memperbarui data yang ditampilkan di bagian metrik, tren, perincian, dan data granular. Kategori tanggal default adalah tanggal pencairan terakhir dan menarik data dari enam bulan terakhir.

Bagian 3: Metrik utama

Bagian dasbor ini menampilkan indikator kinerja utama (KPI) untuk memvisualisasikan perbandingan antara angka pendapatan yang dicairkan dan tidak dicairkan. KPI ditampilkan untuk pendapatan kotor, pendapatan bersih, biaya grosir (jika berlaku), jumlah yang dicairkan, dan jumlah yang tidak dicairkan untuk kriteria filter yang ditentukan.

Bagian 4: Tren

Bagian dasbor ini memberikan tampilan pencairan dan tren jatuh tempo masa lalu untuk rentang tanggal yang ditentukan. Anda dapat melihat tren berdasarkan agregasi tanggal tertentu, seperti berdasarkan hari, bulan, kuartal, atau tahun, untuk mendapatkan wawasan tentang kesehatan AWS Marketplace koleksi Anda. Tampilan tren meliputi yang berikut:

- **Tren pencairan** — Memberikan gambaran tentang jumlah rata-rata hari untuk mencairkan dan pendapatan bersih terkait. Tren mengukur jumlah hari antara tanggal faktur dan tanggal pencairan untuk melaporkan efisiensi pengumpulan. Anda dapat memilih rentang tanggal dari filter agregasi tanggal.
- **Usia pembayaran yang dcairkan** - Memberikan gambaran pendapatan bersih dan hitungan faktur yang dcairkan yang dikategorikan berdasarkan ember piutang penuaan standar (seperti tidak jatuh tempo, 1—30 hari, dan 31 hingga 60 hari). Tren mengukur hari-hari antara tanggal jatuh tempo pembayaran dan tanggal pencairan untuk melaporkan jika pencairan berada dalam ketentuan pembayaran pelanggan.
- **Usia pembayaran yang tidak dcairkan** - Memberikan gambaran pendapatan bersih dan jumlah faktur terbuka dan belum dibayar, yang diatur oleh ember jatuh tempo sebelumnya (seperti tidak jatuh tempo, 1—30 hari, dan 31 hingga 60 hari). Dana yang belum dcairkan mungkin termasuk jumlah yang belum jatuh tempo. Tren mengukur hari-hari antara tanggal hari ini dan tanggal jatuh tempo pembayaran untuk menampilkan piutang yang masuk.

Bagian 5: Kerusakan

Bagian dasbor ini memberi Anda pandangan piutang berdasarkan ID penawaran, judul produk, nama perusahaan pembayar, nama perusahaan pelanggan, nama pengecer (jika mereka berpartisipasi dalam penawaran pribadi mitra saluran), geografi pembayar, dan geografi pelanggan. Gunakan rincian untuk mengukur piutang yang dcairkan terhadap piutang yang tidak dcairkan untuk setiap kategori.

Bagian 6: Data granular

Bagian dasbor ini menunjukkan semua pencairan dan dana yang tidak terkumpul berdasarkan produk, pelanggan, dan detail penawaran.

Note

Faktur yang dibuat sebelum 1 April 2021 mungkin tidak memiliki ID perjanjian terkait, ID penawaran, Akun AWS ID pelanggan, atau nama perusahaan pelanggan.

Untuk informasi tentang cara mengekspor dan mengunduh data dari QuickSight tabel, lihat [Mengekspor data dari visual](#) di QuickSight Panduan Pengguna.

Deskripsi data granular

Kolom	Deskripsi
Tanggal faktur	Tanggal pelanggan ditagih untuk berlangganan produk.
Tanggal jatuh tempo pembayaran	Tanggal jatuh tempo pembayaran dalam format YYYY-MM-DD.
Ketentuan pembayaran	Ketentuan pembayaran AWS faktur pelanggan.
ID Faktur	AWS ID yang ditetapkan ke AWS faktur tempat tagihan ditagih.
ID faktur biaya cantuman	Ketika AWS Marketplace langganan ditransaksikan melalui AWS EMEA SARL, Jepang, atau badan hukum Australia (penjual catatan), operator pasar untuk penjualan (misalnya, AWS EMEA SARL) diharuskan membebankan PPN kepada penjual atas biaya daftar penjual. Untuk transaksi yang berlaku, ID faktur untuk PPN yang dinilai pada biaya listing berbeda dari ID faktur langganan perangkat lunak atau produk.
ID faktur grosir	AWS ID yang ditetapkan ke faktur yang tidak dapat dibayar yang mewakili penjualan antara ISV dan Mitra Saluran dalam penawaran Pribadi Mitra Saluran (CPPO). Untuk penawaran umum dan penawaran Marketplace Private (MPOs) bidang ini akan menjadi 'Tidak berlaku'.
Nama perusahaan pelanggan	Nama akun yang berlangganan produk.
ID Pelanggan Akun AWS	ID akun yang berlangganan produk.

Kolom	Deskripsi
Domain email pelanggan	Domain email yang terkait dengan akun yang berlangganan produk. Misalnya, jika alamat emailnya adalah liu-jie@example.com, entri adalah example.com.
Kota pelanggan	Kota alamat penagihan terkait dengan akun yang berlangganan produk.
Negara atau wilayah pelanggan	Status alamat penagihan yang terkait dengan akun yang berlangganan produk.
Negara pelanggan	Negara alamat penagihan yang terkait dengan akun yang berlangganan produk.
Kode pos pelanggan	Kode pos alamat penagihan terkait dengan akun yang berlangganan produk.
Judul produk	Judul produk.
Nama penawaran	Nama penawaran yang ditentukan penjual.
ID Penawaran	Pengidentifikasi untuk penawaran yang ditandatangani pembeli.
Menawarkan visibilitas	Apakah penawaran itu adalah penawaran kontrak publik, swasta, atau perusahaan.
ID Perjanjian	Referensi umpan data perjanjian unik untuk perjanjian yang ditandatangani antara pengusul dan penerima untuk mulai menggunakan produk.
Tanggal mulai perjanjian	Tanggal langganan produk pelanggan dimulai, diformat sebagai MM-DD-YYYY. Tanggal ini bisa berbeda dari tanggal penerimaan jika ini adalah perjanjian bertanggal masa depan.

Kolom	Deskripsi
Tanggal akhir perjanjian	Tanggal ketika kontrak berakhir, diformat sebagai. MM-DD-YYYY Untuk pay-as-you-go meter/langganan, tanggal ini diatur ke JAN-1-9999.
Tanggal penerimaan perjanjian	Tanggal ketika pelanggan berlangganan produk, diformat sebagai. MM-DD-YYYY
Tanggal akhir periode penggunaan	Tanggal akhir periode penggunaan produk.
Tanggal mulai periode penggunaan	Tanggal mulai periode penggunaan produk.
Status pencairan	Status yang terkait dengan faktur untuk mengonfirmasi bahwa AWS telah mengumpulkan dan mencairkan dana ke rekening bank Anda sejak pencairan sebelumnya. Dana yang dcairkan untuk faktur terkait telah dikumpulkan dan dcairkan. Dana yang tidak dcairkan untuk faktur terkait belum dikumpulkan dan dcairkan.
Tanggal pencairan	Tanggal AWS dimulai pencairan ke bank penjual.
Pencairan ID jejak bank	Untuk pencairan, ID jejak diberikan oleh bank. ID jejak bank dapat digunakan untuk menghubungkan pemberitahuan setoran yang disediakan bank penjual dan laporan ke faktur dalam laporan. AWS Marketplace
Pendapatan kotor	Jumlah yang ditagihkan kepada pelanggan untuk penggunaan atau biaya bulanan produk.
Pengembalian dana kotor	Jumlah total biaya berlangganan dikembalikan kepada pelanggan jika ada pengembalian uang yang diproses selama periode cakupan data.

Kolom	Deskripsi
Biaya listing	Jumlah AWS Marketplace biaya yang akan dikurangkan dari jumlah yang ditagih.
Pengembalian biaya listing	Bagian dari AWS Marketplace biaya dikembalikan jika ada pengembalian uang yang diproses selama periode cakupan data.
Persentase biaya daftar	Persentase AWS Marketplace biaya yang akan dikurangkan dari jumlah yang ditagih.
Pembagian pajak penjual	Jumlah total pajak penjualan dan penggunaan AS yang ditagih untuk transaksi ini.
Pengembalian pajak saham penjual	Jumlah total pajak penjualan dan penggunaan AS dikembalikan untuk transaksi ini jika pengembalian dana diproses.
AWS biaya daftar saham pajak	Dimana AWS sebagai operator pasar diharuskan untuk membebaskan dan memungut PPN atas namanya sendiri atas penjualan yang dilakukan oleh penjual. Jumlah ini akan sesuai dengan jumlah PPN yang dibebankan ke akun AWS Marketplace Penjual Anda.
AWS biaya daftar pengembalian dana bagian pajak	Dimana AWS sebagai operator pasar diharuskan untuk mengembalikan PPN atas namanya sendiri atas penjualan yang dilakukan oleh penjual. Jumlah ini akan sesuai dengan jumlah PPN yang dikembalikan ke akun AWS Marketplace Penjual Anda.

Kolom	Deskripsi
Biaya grosir	Hanya untuk penawaran pribadi mitra saluran. Biaya barang ke pengecer. Misalnya, apa yang dibayar pengecer kepada produsen ketika mereka menjual produk pabrikan. Biaya grosir adalah harga daftar dikalikan dengan persentase diskon.
Pengembalian biaya grosir	Hanya untuk penawaran pribadi mitra saluran. Biaya barang yang dikembalikan dari pengecer.
Pangsa pajak penjual grosir	Pajak atas penjualan antara ISV dan Mitra Saluran dalam penawaran Pribadi Mitra Saluran (CPPO) di mana penjual bertanggung jawab pajak. Untuk penawaran umum dan penawaran Marketplace Private (MPPOs) bidang ini akan menjadi 'Tidak berlaku'.
Pengembalian saham pajak penjual grosir	Pengembalian pajak atas penjualan antara ISV dan Mitra Saluran dalam penawaran Pribadi Mitra Saluran (CPPO) di mana penjual bertanggung jawab. Untuk penawaran umum dan penawaran Marketplace Private (MPPOs) bidang ini akan menjadi 'Tidak berlaku'.
Grosir bagian pajak penjual lainnya	Pajak atas penjualan antara ISV dan Mitra Saluran dalam penawaran Pribadi Mitra Saluran (CPPO) di mana penjual bertanggung jawab pajak. Bidang ini diisi ketika penjual lain yang terlibat dalam transaksi bertanggung jawab. Untuk penawaran umum dan penawaran Marketplace Private (MPPOs) bidang ini akan menjadi 'Tidak berlaku'.

Kolom	Deskripsi
Pengembalian saham pajak penjual grosir lainnya	Pengembalian pajak atas penjualan antara ISV dan Mitra Saluran dalam penawaran Pribadi Mitra Saluran (CPPO) di mana penjual bertanggung jawab. Bidang ini diisi ketika penjual lain yang terlibat dalam transaksi bertanggung jawab. Untuk penawaran umum dan penawaran Marketplace Private (MPPOs) bidang ini akan 'Tidak berlaku'..
Pembagian AWS pajak grosir	Pajak atas penjualan antara ISV dan Mitra Saluran dalam penawaran Pribadi Mitra Saluran (CPPO) yang AWS dikenakan pajak. Untuk penawaran umum dan penawaran Marketplace Private (MPPOs) bidang ini akan menjadi 'Tidak berlaku'.
Pengembalian AWS pajak saham grosir	Pengembalian pajak atas penjualan antara ISV dan Mitra Saluran dalam penawaran Pribadi Mitra Saluran (CPPO) yang AWS dikenakan pajak. Untuk penawaran umum dan penawaran Marketplace Private (MPPOs) bidang ini akan menjadi 'Tidak berlaku'.
Pendapatan bersih penjual	Jumlah total yang ditagih untuk transaksi, setelah dikurangi AWS Marketplace biaya, pengembalian uang, dan pajak penjualan dan penggunaan AS.
Mata Uang	Mata uang transaksi. Misalnya, jika transaksi dalam dolar AS, entri adalah USD.
ID referensi transaksi	Pengidentifikasi unik yang mewakili transaksi , yang dapat Anda gunakan untuk menghubungkan transaksi di seluruh AWS Marketplace laporan.

Kolom	Deskripsi
AWS penjual catatan	<p>Pengenalan entitas bisnis yang memfasilitasi transaksi. Kemungkinan nilainya adalah sebagai berikut:</p> <ul style="list-style-type: none"> • AWS_INC: Pengidentifikasi untuk AWS, Inc. (berbasis di Amerika Serikat) • AWS_EUROPE: Pengidentifikasi untuk AWS EMEA SARL (berbasis di Luksemburg) • AWS Australia Pty Ltd (AWS Australia) • AWS Jepang G.K. • AWS_KOREA
ID otorisasi penjualan kembali	Pengidentifikasi unik untuk peluang terdaftar.
Nama otorisasi penjualan kembali	Nama unik untuk peluang terdaftar.
Deskripsi otorisasi penjualan kembali	Deskripsi yang ditentukan ISV untuk peluang terdaftar.
Nama perusahaan reseller	Nama akun yang membeli produk atau layanan dengan biaya grosir dari ISV untuk dijual kembali ke pelanggan.
ID Pengecer Akun AWS	ID akun yang membeli produk atau layanan dengan biaya grosir dari ISV untuk dijual kembali ke pelanggan.
Nama perusahaan pembayar	Nama akun tempat tagihan ditagih.
ID Pembayar Akun AWS	ID akun tempat tagihan ditagih.
Domain email pembayar	Domain email yang dikaitkan dengan akun tempat tagihan ditagih. Misalnya, jika alamat emailnya adalah liu-jie@example.com, entri adalah example.com.

Kolom	Deskripsi
Kota pembayar	Kota alamat penagihan yang terkait dengan akun yang dikenakan biaya ditagih.
Negara bagian atau wilayah pembayar	Alamat penagihan menyatakan terkait dengan akun tempat tagihan ditagih.
Negara pembayar	Kode negara dua karakter yang terkait dengan akun tempat tagihan ditagih.
Kode pos pembayar	Kode pos alamat penagihan yang terkait dengan akun tempat tagihan ditagih.
ID akun ISV	Pengidentifikasi pemilik produk atau layanan.
Nama perusahaan ISV	Nama bisnis pemilik produk atau layanan.
ID Produk	Pengidentifikasi unik yang ramah untuk produk perangkat lunak.
Status pencairan	Status yang terkait dengan faktur untuk mengonfirmasi bahwa AWS tidak ada dana, sebagian atau seluruh dana ke rekening bank Anda. Status yang mungkin adalah: Disbursed, Sebagian dicairkan, Gagal, Tidak dicairkan.
Pendapatan bersih yang dicairkan	Jumlah total untuk transaksi yang dicairkan kepada penjual. Jika jumlahnya tidak sama dengan "pendapatan bersih penjual", maka ini adalah pembayaran sebagian.
Pendapatan bersih yang belum dicairkan	Jumlah total untuk transaksi yang tidak dicairkan kepada penjual. Jika jumlahnya tidak lain dari nol, sisa saldo terutang oleh pelanggan .

Kolom	Deskripsi
Periode pencairan	Kategori yang menggambarkan kisaran piutang di mana dana dikumpulkan (seperti, tidak jatuh tempo, 1-30 hari, dan 31 hingga 60 hari).

Dasbor perpajakan

Dasbor perpajakan menyediakan visualisasi dan data halus tentang pajak AS dan internasional untuk transaksi di AWS Marketplace

Untuk membuka dasbor ini, masuk ke [Portal Manajemen AWS Marketplace](#), pilih Wawasan, Pemasaran, lalu pilih tab Kinerja daftar.

Topik

- [Segarkan frekuensi dasbor perpajakan](#)
- [Bagian 1: Kontrol](#)
- [Bagian 2: Tanggal filter](#)
- [Bagian 3: Metrik](#)
- [Bagian 4: Tren pajak](#)
- [Bagian 5: Kerusakan](#)
- [Bagian 6: Data granular](#)
- [Kode pembebasan AS](#)
- [Kode pembebasan UE](#)

Segarkan frekuensi dasbor perpajakan

Dasbor diperbarui setiap hari. Jika data transaksi terbaru yang diterima dari sistem hulu tertunda, mungkin ada penundaan 1-2 hari agar data terbaru tercermin di dasbor.

Bagian 1: Kontrol

Bagian dasbor ini menyediakan filter untuk menyempurnakan data perpajakan Anda. Anda juga dapat menambahkan filter ke analisis, seperti rentang tanggal yang ingin Anda sertakan dalam visual. Filter yang dipilih dalam kontrol memperbarui data yang ditampilkan di bagian metrik, tren, perincian, dan data granular.

Untuk informasi selengkapnya tentang pemfilteran, lihat [Memfilter data QuickSight di QuickSight Panduan Pengguna](#).

Deskripsi kontrol

Nama kontrol	Deskripsi
ID Faktur	AWS ID yang ditetapkan untuk biaya yang ditagihkan kepada pelanggan.
ID pelanggan Akun AWS yang dikenakan pajak	AWS ID akun yang dikenakan pajak untuk langganan produk.
AWS penjual catatan	Pengidentifikasi entitas bisnis yang memfasilitasi transaksi.
Negara pelanggan yang dikenai pajak	Kode negara dua karakter yang terkait dengan pelanggan yang dikenakan pajak
Negara atau wilayah pelanggan yang dikenakan pajak	Alamat penagihan negara atau wilayah yang terkait dengan pelanggan yang dikenakan pajak.
Kota pelanggan yang dikenakan pajak	Kota alamat penagihan yang terkait dengan pelanggan yang dikenakan pajak
Kena pajak	Sumber transaksi yang kena pajak, tidak kena pajak, dan pengembalian pajak.
Pihak yang bertanggung jawab pajak	Baik AWS atau Seller. Jika penjual adalah pihak yang bertanggung jawab pajak, pajak dikumpulkan. Jika AWS merupakan pihak yang bertanggung jawab pajak, pajak penjualan dikumpulkan dan dikirimkan oleh AWS. Untuk informasi selengkapnya, lihat Bantuan AWS Marketplace Pajak untuk Penjual . Jika tidak ada pajak yang dikumpulkan, tidak ada nilai yang ditampilkan. Penjual harus menentukan apakah beberapa pajak

Nama kontrol	Deskripsi
	dikumpulkan untuk setiap faktur, karena penjual bertanggung jawab atas pengumpulan pajak.
Jenis pajak	Jenis pajak yang diterapkan pada transaksi. Nilai yang mungkin adalah None, Sales, dan SellerUse Tax registration type , VAT (Pajak Pertambahan Nilai),, CNPJ, IGST. CT

Bagian 2: Tanggal filter

Bagian dasbor ini menyediakan filter untuk memperbaiki catatan berdasarkan tanggal faktur untuk rentang tanggal tertentu. Filter yang dipilih dalam tanggal faktur memperbarui data yang ditampilkan di bagian metrik, tren, perincian, dan data terperinci. Misalnya, jika Anda memilih filter tanggal *Past 30 days*, maka semua bagian akan mencerminkan data yang terkait dengan faktur yang dikenakan pajak dalam 30 hari terakhir.

Bagian 3: Metrik

Bagian dasbor ini menampilkan indikator kinerja utama (KPI) untuk memvisualisasikan metrik yang terkait dengan pajak, termasuk jumlah kena pajak, jumlah tidak kena pajak, jumlah pajak, jumlah faktur yang dikenakan pajak, dan jumlah total transaksi. Anda dapat memperbarui rentang tanggal dengan memperbarui filter tanggal di bagian Filter.

Bagian 4: Tren pajak

Bagian dasbor ini menyediakan tren perpajakan untuk rentang tanggal tertentu. Anda dapat melihat tren berdasarkan agregasi tanggal yang ditentukan—seperti harian,, month-over-month quarter-over-quarter, atau year-over-year —untuk mendapatkan wawasan tentang perpajakan. Informasi tren perpajakan berikut tersedia:

- **Tren pajak - jumlah** - Memberikan gambaran tentang jumlah kena pajak dan pajak untuk periode waktu yang dipilih berdasarkan tanggal faktur.
- **Tren pajak - jumlah faktur** - Memberikan snapshot jumlah faktur untuk periode waktu yang dipilih berdasarkan tanggal faktur.

Bagian 5: Kerusakan

Bagian dasbor ini memberi Anda metrik pajak untuk bisnis Anda berdasarkan ID faktur, jenis pajak, judul produk, dan geografi pelanggan.

Bagian 6: Data granular

Bagian dasbor ini menampilkan data pajak terperinci untuk transaksi organisasi Anda. AWS Marketplace

Deskripsi data granular

Kolom	Deskripsi
ID Faktur	AWS ID yang ditetapkan ke AWS faktur tempat tagihan ditagih. Ini bisa berupa faktur komersial pembeli, faktur grosir penjual atau faktur biaya daftar penjual.
ID faktur biaya daftar	AWS ID yang ditetapkan ke faktur tempat biaya daftar penjual dan PPN ditagih kepada penjual. Berlaku ISVs hanya untuk.
ID item baris	Pengidentifikasi unik untuk item baris. Transaksi pengembalian dana memiliki ID item baris yang sama dengan transaksi pajak forward mereka.
ID tagihan pelanggan	ID tagihan pelanggan
Pihak yang bertanggung jawab pajak	Baik AWS atau Seller. Jika penjual adalah pihak yang bertanggung jawab pajak, pajak dikumpulkan. Jika AWS merupakan pihak yang bertanggung jawab pajak, pajak penjualan dikumpulkan dan dikirimkan oleh AWS. Untuk informasi selengkapnya, lihat Bantuan AWS Marketplace Pajak untuk Penjual . Jika tidak ada pajak yang dikumpulkan, tidak ada nilai yang ditampilkan. Penjual

Kolom	Deskripsi
	<p>harus menentukan apakah beberapa pajak dikumpulkan untuk setiap faktur, karena penjual bertanggung jawab atas pengumpulan pajak.</p>
Kode jenis transaksi	<p>Jenis kode transaksi. Nilai-nilai meliputi:</p> <ul style="list-style-type: none"> • AWS— Transaksi pajak forward. • REFUND— Pengembalian dana penuh atau sebagian. • TAXONLYREFUND — Pengembalian uang hanya pajak. <p>Transaksi pengembalian dana membagikan ID item baris dengan transaksi forward asli mereka.</p>
ID Produk	<p>Pengidentifikasi unik yang ramah untuk produk perangkat lunak.</p>
Judul produk	<p>Nama produk yang dibeli.</p>
Kode pajak produk	<p>Kode standar untuk mengidentifikasi properti pajak untuk suatu produk. Anda memilih properti saat Anda membuat atau memodifikasi produk.</p>
Tanggal faktur	<p>Tanggal pelanggan ditagih dan dikenakan pajak untuk langganan produk.</p>
ID pelanggan Akun AWS yang dikenakan pajak	<p>Akun AWS ID pelanggan yang dikenakan pajak.</p>
Negara pelanggan yang dikenai pajak	<p>Kode negara dua karakter yang terkait dengan pelanggan yang dikenakan pajak.</p>

Kolom	Deskripsi
Negara atau wilayah pelanggan yang dikenakan pajak	Alamat penagihan negara atau wilayah yang terkait dengan pelanggan yang dikenakan pajak.
Kota pelanggan yang dikenakan pajak	Kota alamat penagihan yang terkait dengan pelanggan yang dikenakan pajak.
Kode pos pelanggan yang dikenakan pajak	Kode pos yang digunakan untuk perhitungan pajak.
Jenis pajak	Jenis pajak yang diterapkan pada transaksi . Nilai yang mungkin adalahNone,Sales, danSellerUse , Tax registration type,VAT,CNPJ,IGST,CT.
Tingkat yurisdiksi	Tingkat yurisdiksi alamat yang digunakan untuk lokasi pajak. Nilai yang mungkin adalahState,County,City, danDistrict.
Yurisdiksi pajak	Nama yurisdiksi pajak.
Jenis pajak harga tampilan	Jenis pajak untuk harga yang muncul untuk pelanggan. Semua AWS Marketplace penawaran eksklusif.
Tarif yurisdiksi pajak	Tarif pajak yang diterapkan di tingkat yurisdiksi.
Jumlah pajak	Pajak yang dibebankan di tingkat yurisdiksi.
Mata Uang	Mata uang yang dikenakan pajak. Misalnya, jika transaksi dalam dolar AS, entri adalah USD.
Jumlah pajak legal	Jumlah pajak yang ditunjukkan dalam mata uang hukum yang dibebankan pada tingkat yuridisi.

Kolom	Deskripsi
Mata uang pajak legal	Mata uang yang ditunjukkan dalam mata uang hukum pajak yang dibebankan.
Kode alasan perhitungan pajak	Apakah transaksi tersebut kena pajak, tidak kena pajak, dikecualikan, atau dinilai nol, yang diatur oleh tingkat yurisdiksi.
Tanggal digunakan untuk perhitungan pajak	Tanggal yang digunakan untuk menghitung pajak atas transaksi.
ID sertifikat pembebasan pelanggan	ID sertifikat sertifikat pembebasan.
Domain ID sertifikat pembebasan pelanggan	Di mana sertifikat disimpan dalam sistem Amazon.
Tingkat sertifikat pembebasan pelanggan	Tingkat yurisdiksi yang memberikan pengecualian.
Kode pembebasan pelanggan	Kode yang menentukan pengecualian. Misalnya, RESALE.
ID referensi transaksi	Pengidentifikasi unik untuk transaksi yang membantu Anda menghubungkan transaksi di seluruh AWS Marketplace laporan.
AWS penjual catatan	Pengidentifikasi entitas bisnis yang memfasilitasi transaksi.
ID faktur pembeli	AWS ID yang ditetapkan ke AWS faktur yang dibebankan ditagih kepada pembeli. Penjual dapat mengaitkan faktur grosir atau faktur biaya daftar mereka dengan biaya pembeli yang sesuai.

Note

Jumlah pajak total dan total biaya pembeli telah pindah ke bagian AWS pajak dan bidang pendapatan kotor di [dasbor pendapatan yang ditagih](#).

Kode pembebasan AS

Kode pengecualian	Deskripsi
501C	Organisasi 501C
AP	Produksi Pertanian
CO	Organisasi Amal
LANGSUNG	Izin Pembayaran Langsung
DISTSBT	Distributor
DP	Perserikatan Bangsa-Bangsa/Diplomat
DV	Veteran Cacat
EDI	Institusi Pendidikan
FG	Pemerintah Federal
HCP	Penyedia Layanan Kesehatan
HO	Rumah Sakit (Nirlaba atau Negara Bagian)
INSUR	Asuransi
IPM	IPM
LB	Perpustakaan
MPU	Pembebasan Beberapa Titik Penggunaan; Pengecualian Produk/Perangkat Lunak Digital

Kode pengecualian	Deskripsi
TA	Penduduk asli Amerika
NAI	Individu Asli Amerika
NP	Organisasi Nirlaba
NPR	Organisasi Keagamaan Nirlaba
OT	Lainnya
DIJUAL KEMBALI	Pengecer
SBE	Pembebasan Usaha Kecil
SLG	Pemerintah Negara/Lokal

Kode pembebasan UE

Kode pengecualian	Deskripsi
SpecialOrg	Organizations yang dibebaskan dari dikenakan PPN.
SpecialZone	Area yang bebas pajak.
SplitPayment	Pembeli melakukan pembayaran PPN yang tercantum pada faktur langsung ke otoritas pajak.

Dasbor untuk operasi penjualan

AWS Marketplace menyediakan beberapa dasbor untuk membantu Anda melacak data penjualan Anda.

Topik

- [Dasbor perjanjian dan pembaruan](#)

- [Dasbor penggunaan](#)

Dasbor perjanjian dan pembaruan

Dasbor ini memberikan informasi tentang perjanjian dan pembaruan dalam waktu 24 jam setelah penandatanganan perjanjian. AWS Marketplace Dasbor ini dapat membantu Anda melacak langganan yang kedaluwarsa.

Untuk membuka dasbor, mulai Portal Manajemen AWS Marketplace, pilih tab Wawasan, lalu pilih Operasi penjualan.

Topik

- [Bagian 1: Kontrol](#)
- [Bagian 2: Pilih kategori tanggal](#)
- [Bagian 3: Metrik](#)
- [Bagian 4: Tren berlangganan](#)
- [Bagian 5: Kerusakan](#)
- [Bagian 6: Data granular](#)

Bagian 1: Kontrol

Bagian ini menyediakan filter untuk menyempurnakan perjanjian dan data perpanjangan Anda. Anda dapat memilih dari filter berikut.

Note

Untuk informasi selengkapnya tentang pemfilteran, lihat [Memfilter data QuickSight di QuickSight](#) Panduan Pengguna.

Deskripsi kontrol

Nama kontrol	Deskripsi
ID Pelanggan Akun AWS	ID akun yang berlangganan produk.
Nama perusahaan pelanggan	Nama akun yang berlangganan produk.

Nama kontrol	Deskripsi
ID Penawaran	Pengidentifikasi untuk penawaran yang ditandatangani pembeli.
Menawarkan visibilitas	Apakah penawaran itu adalah penawaran kontrak publik, swasta, atau perusahaan.
ID Perjanjian	Referensi umpan data perjanjian unik untuk perjanjian yang ditandatangani antara pengusul dan penerima untuk mulai menggunakan produk.
Bendera CPPO	Bendera ya/tidak yang menunjukkan apakah perjanjian dibuat menggunakan penawaran pribadi mitra saluran. Jika ya, penjual rekaman adalah mitra saluran. Jika tidak, penjual catatan adalah produsen perangkat lunak (vendor perangkat lunak independen).
ID AWS akun ISV	Akun pemilik produk atau layanan.
Nama perusahaan ISV	Nama bisnis pemilik produk atau layanan.
ID produk lama	Pengidentifikasi unik warisan untuk produk perangkat lunak.
Judul produk	Judul produk.
ID otorisasi penjualan kembali	ID otorisasi penjualan kembali yang diberikan oleh ISV kepada mitra saluran.
Nama otorisasi penjualan kembali	Nama otorisasi penjualan kembali yang diberikan oleh ISV kepada mitra saluran.
ID Pengecer Akun AWS	ID akun yang membeli produk atau layanan secara grosir dari ISV untuk dijual kembali ke pelanggan.

Nama kontrol	Deskripsi
Nama perusahaan reseller	Nama bisnis akun pengecer yang berwenang untuk menjual produk produsen perangkat lunak.
Negara pelanggan	Kode negara dua karakter yang terkait dengan akun berlangganan produk.
Negara atau wilayah pelanggan	Status alamat penagihan atau wilayah yang terkait dengan akun berlangganan produk.
Kota pelanggan	Kota alamat penagihan yang terkait dengan akun berlangganan produk.

Bagian 2: Pilih kategori tanggal

Bagian dasbor ini menyediakan filter untuk menyempurnakan catatan berdasarkan tanggal akhir perjanjian, periode akhir perjanjian, atau tanggal mulai perjanjian untuk rentang tanggal yang ditentukan. Filter yang dipilih dalam kategori tanggal yang dipilih memperbarui data yang ditampilkan di bagian metrik, tren, perincian, dan data granular.

Untuk informasi tentang perjanjian yang baru ditandatangani, Anda dapat memfilter berdasarkan tanggal mulai perjanjian. Jika Anda tertarik untuk mempelajari tentang perpanjangan perjanjian, Anda dapat memfilter menggunakan tanggal akhir perjanjian.

Bagian 3: Metrik

Bagian dasbor ini menampilkan indikator kinerja utama (KPI) untuk memvisualisasikan perbandingan data perjanjian utama. KPI ditampilkan untuk jumlah perjanjian aktif dan jumlah perjanjian yang berakhir. Anda dapat memperbarui rentang tanggal dengan memperbarui kriteria tanggal di kolom Pilih kategori tanggal.

Bagian 4: Tren berlangganan

Bagian dasbor ini menyediakan tren berlangganan untuk rentang tanggal tertentu. Anda dapat melihat tren berdasarkan agregasi tanggal yang ditentukan—seperti harian,, month-over-month quarter-over-quarter, atau year-over-year —untuk mendapatkan wawasan tentang perjanjian. Informasi tren perjanjian dan pembaruan tersedia pada tingkat agregat.

- **Tren berlangganan** — Memberikan gambaran tentang jumlah perjanjian aktif dan jumlah perjanjian yang berakhir berdasarkan filter agregasi tanggal yang dipilih dan filter kategori tanggal yang dipilih. Untuk melihat tren langganan untuk tanggal mulai perjanjian, Anda dapat memilih Tanggal mulai Perjanjian di filter kategori Pilih tanggal. Untuk melihat tren langganan untuk tanggal akhir perjanjian, pilih Tanggal akhir perjanjian di filter kategori Pilih tanggal.
- **Rincian pembaruan** - Menyediakan snapshot pembaruan yang telah berakhir atau akan berakhir dalam jumlah hari tertentu. Anda dapat melihat rincian perpanjangan untuk beberapa periode waktu, seperti perjanjian yang berakhir satu hingga 30 hari yang lalu, 31 hingga 60 hari yang lalu, atau lebih dari 60 hari yang lalu. Anda juga dapat memfilter perjanjian yang akan berakhir dalam nol hingga 30 hari, 31 hingga 60 hari, dan seterusnya.

Bagian 5: Kerusakan

Bagian dasbor ini memberi Anda metrik untuk bisnis Anda di seluruh nama perusahaan untuk pelanggan aktif ISVs, dan pengecer. Anda dapat melihat jumlah perjanjian IDs, perjanjian berakhir, penawaran IDs IDs, perjanjian yang diperbarui IDs, dan pelanggan.

Bagian 6: Data granular

Bagian dasbor ini menunjukkan data terperinci untuk perjanjian, penawaran, produk, pengusul perjanjian, pelanggan, otorisasi penjualan kembali, pengecer, dan. ISVs

Note

Untuk informasi selengkapnya tentang cara mengekspor dan mengunduh data dari QuickSight tabel, lihat [Mengekspor data dari visual](#) di QuickSight Panduan Pengguna.

Deskripsi data granular

Kolom	Deskripsi
Nama perusahaan pelanggan	Nama perusahaan pelanggan.
ID AWS akun pelanggan	ID akun yang berlangganan produk.
Domain email pelanggan	Domain email yang terkait dengan akun yang berlangganan produk. Misalnya, jika alamat

Kolom	Deskripsi
	emailnya adalah liu-jie@example.com, entri adalah example.com.
Negara pelanggan	Negara alamat penagihan yang terkait dengan akun yang berlangganan produk.
Negara atau wilayah pelanggan	Status alamat penagihan yang terkait dengan akun yang berlangganan produk.
Kota pelanggan	Kota alamat penagihan terkait dengan akun yang berlangganan produk.
Kode pos pelanggan	Kode pos alamat penagihan terkait dengan akun yang berlangganan produk.
Judul produk	Judul produk.
ID Produk	Pengidentifikasi unik yang ramah untuk produk perangkat lunak.
ID produk lama	Pengidentifikasi unik warisan untuk produk perangkat lunak.
Nama penawaran	Nama penawaran yang ditentukan penjual.
ID Penawaran	Pengidentifikasi untuk penawaran yang ditandatangani pembeli.
Menawarkan visibilitas	Apakah penawaran itu adalah penawaran kontrak publik, swasta, atau perusahaan.
ID Perjanjian	Referensi umpan data perjanjian unik untuk perjanjian yang ditandatangani antara pengusul dan penerima untuk mulai menggunakan produk.

Kolom	Deskripsi
Tanggal mulai perjanjian	Tanggal langganan produk pelanggan dimulai, diformat sebagai MM-DD-YYYY. Tanggal ini bisa berbeda dari tanggal penerimaan jika ini adalah perjanjian bertanggal masa depan.
Tanggal penerimaan perjanjian	Tanggal ketika pelanggan berlangganan produk, diformat sebagai MM-DD-YYYY
Tanggal akhir perjanjian	Tanggal ketika kontrak berakhir, diformat sebagai MM-DD-YYYY Untuk pay-as-you-go meter/langganan, tanggal ini diatur ke JAN-1-9999.

Kolom	Deskripsi
Status	<p>Status perjanjian saat ini. Status yang didukung meliputi:</p> <ul style="list-style-type: none">• Aktif — Beberapa atau semua ketentuan perjanjian berlaku.• Diakhiri - Perjanjian berakhir sebelum tanggal akhir yang disepakati sebelumnya karena peristiwa penghentian yang diprakarsai AWS. Paling sering kegagalan pembayaran.• Dibatalkan — Akseptor memilih untuk mengakhiri perjanjian sebelum tanggal berakhirnya.• Kedaluwarsa - Perjanjian berakhir pada tanggal akhir yang telah disepakati sebelumnya.• Diganti - Perjanjian diganti menggunakan penawaran pengganti.• Diperpanjang - Perjanjian diperbarui menjadi perjanjian baru menggunakan fungsionalitas seperti perpanjangan otomatis.• Diarsipkan — Perjanjian telah berakhir; Alasan pasti untuk berakhirnya, Dihentikan, Dibatalkan, atau Kedaluwarsa, tidak ditentukan.• Rolled back — Revisi perjanjian telah dibatalkan karena kesalahan. Revisi sebelumnya sekarang aktif. Hanya berlaku untuk revisi perjanjian yang tidak aktif.• Superseded — Revisi perjanjian tidak lagi aktif dan revisi perjanjian lainnya sekarang aktif. Hanya berlaku untuk revisi perjanjian yang tidak aktif.

Kolom	Deskripsi
	<ul style="list-style-type: none"> • Rolled back — Revisi perjanjian telah dibatalkan karena kesalahan. Revisi sebelumnya sekarang aktif. Hanya berlaku untuk revisi perjanjian yang tidak aktif.
Estimasi nilai perjanjian	Perkiraan biaya perjanjian atau dikenal sebagai “Total nilai kontrak” ATAU “Jumlah pembelian”. Ini berlaku untuk jenis Produk berikut: SaaS, Layanan Profesional, dan Server, dan jenis harga Kontrak atau Tahunan.
Kode mata uang	Penawaran harga mata uang terkait dengan perkiraan biaya perjanjian.
Nama perusahaan reseller	Nama akun yang membeli produk atau layanan dengan biaya grosir dari ISV untuk dijual kembali ke pelanggan.
ID AWS akun pengecer	ID akun yang membeli produk atau layanan dengan biaya grosir dari ISV untuk dijual kembali ke pelanggan.
ID otorisasi penjualan kembali	Pengidentifikasi unik untuk peluang terdaftar.
Nama otorisasi penjualan kembali	Nama unik untuk peluang terdaftar.
Deskripsi otorisasi penjualan kembali	Deskripsi yang ditentukan ISV untuk peluang terdaftar.
Bendera CPPO	Bidang yang dihitung di QuickSight.

Kolom	Deskripsi
Niat perjanjian	<p>Mengisi jika perjanjian saat ini adalah perpanjangan atau peningkatan dari perjanjian sebelumnya yang disediakan di kolom <code>agreement_id</code> sebelumnya. Nilai yang mungkin termasuk:</p> <ul style="list-style-type: none">• Upgrade: Untuk penawaran pribadi, upgrade adalah istilah yang luas untuk setiap modifikasi (upgrade, downgrade, amandemen, perluas, perpanjang). Untuk penawaran umum, peningkatan adalah perjanjian khusus untuk mengubah persyaratan (menggunakan penawaran yang berbeda) yang menghasilkan perjanjian baru, dan perjanjian khusus diarsipkan.• Perbarui: Hanya untuk penawaran umum. Ketika perjanjian perpanjangan memiliki tanggal mulai sebagai tanggal akhir perjanjian tertentu dan ID penawaran tidak berubah tetapi ID perjanjian berubah. Misalnya, perjanjian A telah menjadi perjanjian B.• Perpanjangan otomatis: Hanya untuk penawaran umum. Ketika perjanjian baru dibuat ketika perjanjian saat ini berakhir. Pembeli telah mengaktifkan perpanjangan otomatis.• Baru: Pembeli telah menerima persyaratan (menggunakan penawaran) dan berlangganan perjanjian baru.
ID perjanjian sebelumnya	Referensi perjanjian jika perjanjian ini diperbarui atau ditingkatkan.

Kolom	Deskripsi
ID perjanjian berikutnya	ID Perjanjian perjanjian berikutnya jika perjanjian saat ini diperbarui atau ditingkatkan.
ID Penawaran sebelumnya	ID Penawaran yang terkait dengan perjanjian sebelumnya jika perjanjian saat ini diperbarui atau ditingkatkan.
ID penawaran berikutnya	ID Penawaran yang terkait dengan perjanjian berikutnya jika perjanjian saat ini diperbarui atau ditingkatkan.
Tanggal penerimaan ID perjanjian berikutnya	Tanggal ketika perjanjian berikutnya diterima oleh pembeli.
ID akun pengusul AWS	Pengidentifikasi pemilik produk atau layanan.
Nama perusahaan pengusul	Nama bisnis pemilik produk atau layanan.
ID AWS akun ISV	AWS Pengidentifikasi penjual.
Nama perusahaan ISV	Nama bisnis penjual.
Perjanjian berakhir (dalam beberapa hari)	Bidang yang dihitung di Quicksight. Jumlah hari tersisa untuk kesepakatan berakhir mulai hari ini.
Periode akhir perjanjian	Bidang yang dihitung di QuickSight. Periode waktu di mana perjanjian akan berakhir.

Dasbor penggunaan

Dasbor Penggunaan menyediakan visualisasi dan data berbutir halus untuk pelanggan yang menggunakan SaaS dan produk berbasis penggunaan server. AWS Marketplace Penjual dapat menggunakan dasbor ini untuk melacak konsumsi pelanggan di seluruh produk berbasis penggunaan untuk membuat keputusan tentang dukungan produk, harga, konversi dari penawaran publik ke pribadi, dan penghentian produk. Dasbor menyediakan data dari 6 bulan terakhir, yang merupakan jendela bergulir.

Untuk membuka dasbor, masuk ke [Portal Manajemen AWS Marketplace](#), pilih Wawasan, pilih Operasi penjualan, lalu pilih tab Penggunaan.

Note

Dasbor ini menampilkan penggunaan untuk semua tombol dimensi yang ditentukan pada saat pembuatan penawaran. Misalnya, untuk produk Amazon Machine Image (AMI), kunci dimensi adalah tipe instance, dan semua jenis instans yang ditentukan dalam penawaran akan muncul di dasbor ini, bahkan ketika dihargai \$0. Untuk melihat penggunaan produk untuk dimensi yang tidak ditentukan pada saat pembuatan penawaran, pertimbangkan untuk menerbitkan ulang produk untuk menyertakan dimensi yang Anda butuhkan.

Topik

- [Segarkan frekuensi dasbor penggunaan](#)
- [Bagian 1: Kontrol](#)
- [Bagian 2: Filter](#)
- [Bagian 3: Metrik](#)
- [Bagian 4: Tren](#)
- [Bagian 5: Kerusakan](#)
- [Bagian 6: Data granular](#)

Segarkan frekuensi dasbor penggunaan

Dasbor diperbarui setiap hari pada pukul 4 sore PST (tengah malam UTC). Perhatikan bahwa data penggunaan diterima dari sumber data hulu dan mungkin mengalami penundaan, Anda dapat merujuk ke tanggal penggunaan dan tanggal penggunaan yang dilaporkan untuk kejelasan tentang kapan penggunaan terjadi dibandingkan dengan saat dilaporkan di dasbor.

Bagian 1: Kontrol

Bagian dasbor ini menyediakan filter untuk menyempurnakan data penggunaan Anda. Misalnya, Anda dapat memilih dari filter berikut.

Note

Untuk informasi selengkapnya tentang pemfilteran, lihat [Memfilter data QuickSight di QuickSight](#) Panduan Pengguna.

Deskripsi kontrol

Nama kontrol	Deskripsi
Nama perusahaan pengguna akhir	Nama akun yang menggunakan produk.
Akun AWS ID pengguna akhir	ID akun yang menggunakan produk.
Negara pengguna akhir	Kode negara dua karakter yang terkait dengan akun yang menggunakan produk.
Judul produk	Judul produk.
Kode produk	Kode produk hak yang ada digunakan untuk mengukur produk. Nilai ini juga digunakan untuk menggabungkan data dengan laporan, atau untuk mereferensikan apa yang disediakan dalam Layanan AWS Marketplace Pengukuran.
ID Penawaran	Pengidentifikasi untuk penawaran yang ditandatangani pembeli.
Menawarkan visibilitas	Apakah penawaran itu adalah penawaran kontrak publik, swasta, atau perusahaan.
ID Perjanjian	Referensi umpan data perjanjian unik untuk perjanjian yang ditandatangani antara pengusul dan penerima untuk mulai menggunakan produk.
Kunci dimensi	Jenis sumber daya yang terkait dengan penggunaan produk. Kunci dimensi berlaku

Nama kontrol	Deskripsi
	untuk SaaS dan produk berbasis penggunaan server.
Nama perusahaan pelanggan	Nama akun yang berlangganan produk.
ID Pelanggan Akun AWS	ID akun yang berlangganan produk.
Negara pelanggan	Kode negara dua karakter yang terkait dengan akun berlangganan produk.
Nama perusahaan reseller	Nama akun pengecer yang berwenang untuk menjual produk produsen produk.
ID Pengecer Akun AWS	ID akun yang membeli produk atau layanan secara grosir dari ISV untuk dijual kembali ke pelanggan.
ID otorisasi penjualan kembali	ID akun yang membeli produk atau layanan secara grosir dari ISV untuk dijual kembali ke pelanggan.
Bendera CPPO	Bendera ya/tidak yang menunjukkan apakah perjanjian dibuat menggunakan penawaran pribadi mitra saluran. Jika ya, penjual rekaman adalah mitra saluran. Jika tidak, penjual rekaman adalah produsen produk (vendor perangkat lunak independen).

Bagian 2: Filter

Bagian dasbor ini menyediakan filter untuk menyempurnakan catatan berdasarkan tanggal penggunaan. Nilai yang dipilih dalam filter ini memperbarui data yang ditampilkan di bagian metrik, tren, perincian, dan data granular. Pilihan default adalah menarik data untuk penggunaan 6 bulan terakhir.

Bagian 3: Metrik

Bagian dasbor ini menampilkan indikator kinerja utama (KPI) untuk memvisualisasikan metrik yang terkait dengan konsumsi: perkiraan unit penggunaan, pelanggan dengan penggunaan, dan produk dengan penggunaan. Anda dapat memperbarui rentang tanggal dengan memperbarui kriteria tanggal penggunaan di bagian filter. Perhatikan bahwa metrik kunci menampilkan data untuk semua jenis unit.

Bagian 4: Tren

Bagian dasbor ini menyediakan tren penggunaan untuk rentang tanggal tertentu. Anda dapat melihat tren dengan agregasi tanggal tertentu, seperti harian, month-over-month quarter-over-quarter, atau year-over-year untuk mendapatkan wawasan tentang penggunaan. Anda juga dapat memilih jenis unit penggunaan untuk melihat tren penggunaannya secara grafis.

Bagian 5: Kerusakan

Bagian dasbor ini memberi Anda perkiraan metrik penggunaan untuk bisnis Anda di seluruh nama perusahaan, judul produk, kunci dimensi, dan penawaran IDs untuk jenis unit yang dipilih. Anda juga dapat memilih jumlah entri yang akan dilihat.

Bagian 6: Data granular

Bagian dasbor ini menunjukkan data terperinci untuk penggunaan, penawaran, produk, pelanggan, pembayar, pengguna akhir, otorisasi penjualan kembali, pengecer, dan vendor perangkat lunak independen (). ISVs Perhatikan bahwa tabel data granular menampilkan data untuk semua tipe unit.

Pendapatan harus dipertimbangkan diperkirakan sampai penagihan diselesaikan pada akhir bulan. Faktur berbasis penggunaan disajikan kepada pembeli pada hari kedua atau ketiga bulan berikutnya untuk penggunaan bulan sebelumnya (misalnya, pelanggan yang diukur untuk penggunaan antara 11/1 dan 11/30 akan diberikan faktur untuk penggunaan pada 12/2 atau 12/3). Penggunaan terukur mungkin tiba di dasbor ini beberapa hari setelah tanggal penggunaan aktual, sehingga tanggal penggunaan dan tanggal penggunaan yang dilaporkan mungkin berbeda. Ini berarti Anda mungkin perlu mengunjungi dasbor setiap hari sepanjang bulan untuk up-to-date melacak di bulan berjalan. Untuk informasi penagihan pelanggan otoritatif, lihat [dasbor Pendapatan yang ditagih](#) di tab Operasi keuangan.

Note

Untuk informasi selengkapnya tentang pemfilteran, lihat [Mengekspor data dari visual di Panduan Pengguna. QuickSight](#)

Deskripsi data granular

Kolom	Deskripsi
Tanggal penggunaan	Tanggal konsumsi produk pelanggan.
Tanggal penggunaan yang dilaporkan	Tanggal konsumsi produk pelanggan muncul di dasbor wawasan.
Nama perusahaan pengguna akhir	Nama akun yang menggunakan produk.
ID AWS akun pengguna akhir	ID akun yang menggunakan produk.
Domain email pengguna akhir	Domain email yang terkait dengan akun yang menggunakan produk. Misalnya, jika alamat emailnya adalah abc@example.com, entri adalah example.com.
Kota pengguna akhir	Kota yang terkait dengan akun yang menggunakan produk.
Status atau wilayah pengguna akhir	Negara bagian atau wilayah yang terkait dengan akun yang menggunakan produk.
Negara pengguna akhir	Kode negara dua karakter yang terkait dengan akun yang menggunakan produk.
Kode pos pengguna akhir	Kode pos alamat penagihan terkait dengan akun yang menggunakan produk.
Judul produk	Judul produk.
ID produk lama	Pengidentifikasi unik warisan untuk produk.

Kolom	Deskripsi
ID Produk	Pengidentifikasi unik yang ramah untuk produk.
Kode produk	Kode produk hak yang ada digunakan untuk mengukur produk. Nilai ini juga digunakan untuk menggabungkan data dengan laporan, atau untuk mereferensikan apa yang disediakan AWS Marketplace Metering Service.
ID Penawaran	Pengidentifikasi untuk penawaran yang ditandatangani pembeli.
Nama penawaran	Nama penawaran yang ditentukan penjual.
Menawarkan visibilitas	Apakah penawaran itu adalah penawaran kontrak publik, swasta, atau perusahaan.
ID Perjanjian	Referensi umpan data perjanjian unik untuk perjanjian yang ditandatangani antara pengusul dan penerima untuk mulai menggunakan produk.
Tanggal penerimaan perjanjian	Cap tanggal waktu di UTC saat pelanggan berlangganan produk.
Tanggal mulai perjanjian	Cap waktu tanggal di UTC saat langganan produk pelanggan dimulai. Tanggal ini bisa berbeda dari tanggal penerimaan jika ini adalah perjanjian bertanggal masa depan.
Tanggal akhir perjanjian	Tanggal di UTC ketika kontrak berakhir. Untuk pay-as-you-go meter/langganan, tanggal ini diatur ke 1 Jan 9999 12:00 AM.
Kunci dimensi	Jenis sumber daya yang terkait dengan penggunaan produk. Kunci dimensi berlaku untuk SaaS dan produk berbasis penggunaan server.

Kolom	Deskripsi
Wilayah	Wilayah tempat pembeli menyebarkan EC2 instans Amazon.
Perkiraan penggunaan	Jumlah penggunaan yang dicatat untuk produk.
Jenis unit penggunaan	Jenis unit yang penggunaannya dicatat.
Estimasi pendapatan	Pendapatan dari penggunaan produk. Pendapatan harus dipertimbangkan diperkirakan sampai penagihan diselesaikan pada akhir bulan. Faktur berbasis penggunaan disajikan kepada pembeli pada tanggal kedua atau ketiga bulan berikutnya untuk penggunaan bulan sebelumnya.
Mata Uang	Mata uang transaksi. Misalnya, jika transaksi dalam dolar AS, entri adalah USD.
Nama perusahaan pelanggan	Nama akun yang berlangganan produk.
ID Pelanggan Akun AWS	ID akun yang berlangganan produk.
Domain email pelanggan	Domain email yang terkait dengan akun yang berlangganan produk. Misalnya, jika alamat emailnya adalah abc@example.com, entri adalah example.com.
Kota pelanggan	Kota alamat penagihan terkait dengan akun yang berlangganan produk.
Negara atau wilayah pelanggan	Status alamat penagihan yang terkait dengan akun yang berlangganan produk.
Negara pelanggan	Negara alamat penagihan yang terkait dengan akun yang berlangganan produk.

Kolom	Deskripsi
Kode pos pelanggan	Kode pos alamat penagihan terkait dengan akun yang berlangganan produk.
Nama perusahaan pembayar	Nama akun yang membayar produk.
ID Pembayar Akun AWS	ID akun yang membayar produk.
Domain email pembayar	Domain email yang terkait dengan akun yang membayar produk. Misalnya, jika alamat emailnya adalah abc@example.com, entri adalah example.com.
Kota pembayar	Kota alamat penagihan terkait dengan akun yang membayar produk.
Negara bagian atau wilayah pembayar	Status alamat penagihan terkait dengan akun yang membayar produk.
Negara pembayar	Negara alamat penagihan yang terkait dengan akun yang membayar produk.
Kode pos pembayar	Kode pos alamat penagihan terkait dengan akun yang membayar produk.
Nama perusahaan reseller	Nama akun yang membeli produk atau layanan dengan biaya grosir dari ISV untuk dijual kembali ke pelanggan.
ID Pengecer Akun AWS	ID akun yang membeli produk atau layanan dengan biaya grosir dari ISV untuk dijual kembali ke pelanggan.
ID otorisasi penjualan kembali	Pengidentifikasi unik untuk peluang penjualan kembali terdaftar.
Nama otorisasi penjualan kembali	Nama unik untuk peluang penjualan kembali terdaftar.

Kolom	Deskripsi
Deskripsi otorisasi penjualan kembali	Deskripsi untuk peluang penjualan kembali terdaftar.
Bendera CPPO	Bendera ya/tidak yang menunjukkan apakah perjanjian dibuat menggunakan penawaran pribadi mitra saluran. Jika ya, penjual rekaman adalah mitra saluran. Jika tidak, penjual rekaman adalah produsen produk (ISV).
Nama perusahaan ISV	Nama pemilik produk atau layanan.
ID ISV Akun AWS	Pengidentifikasi pemilik produk atau layanan.

Dasbor untuk pemasaran

AWS Marketplace menyediakan beberapa dasbor untuk membantu Anda melacak data pemasaran Anda.

Topik

- [Dasbor perjanjian pelanggan](#)
- [Daftar dasbor kinerja](#)
- [Dasbor kinerja pencarian](#)
- [Memahami Beli dengan AWS dasbor](#)

Dasbor perjanjian pelanggan

Dasbor Perjanjian Pelanggan memberikan gambaran umum tentang perjanjian dan pelanggan yang berlangganan produk Anda AWS Marketplace. Dasbor menyediakan data tentang perjanjian baru dan aktif Anda, ditambah tren dan profil pelanggan.

Note

- Untuk membuka kunci dasbor ini, Anda harus mendaftarkan program [AWS Marketplace Penjual Prime](#).

- Untuk membuka dasbor ini, masuk ke [Portal Manajemen AWS Marketplace](#), pilih Wawasan, Pemasaran, lalu pilih tab Perjanjian pelanggan.

Untuk informasi selengkapnya tentang menggunakan AWS Marketplace dasbor, lihat [Dasbor penjual](#), sebelumnya di bagian ini.

Topik

- [Bagian 1: Filter](#)
- [Bagian 2: Filter tanggal menyelam dalam](#)
- [Bagian 3: Perjanjian penawaran publik dan pribadi](#)
- [Bagian 4: Metrik pelanggan](#)

Bagian 1: Filter

Anda dapat menggunakan filter berikut untuk menyempurnakan data perjanjian Anda.

Nama kontrol	Deskripsi
Judul produk	Judul produk.
Filter tanggal	Termasuk data selama 30, 60, dan 90 hari terakhir, trailing 12 bulan (TTM), dan year to date (YTD). Pilih kustom untuk menentukan tanggal mulai dan berakhir tertentu.
Industri pelanggan	Industri yang AWS mendefinisikan untuk pelanggan yang berlangganan produk penjual.
Segmen pelanggan	Segmen yang AWS mendefinisikan untuk pelanggan yang berlangganan produk penjual.

Untuk informasi selengkapnya tentang pemfilteran, lihat [Memfilter data QuickSight di QuickSight Panduan Pengguna Amazon](#).

Bagian 2: Filter tanggal menyelam dalam

Filter tanggal ini berlaku untuk semua metrik pada dasbor perjanjian pelanggan.

Misalnya, saat Anda menggunakan nilai year-to-date filter default, metrik berikut akan muncul:

Metrik	Deskripsi
Jumlah perjanjian penawaran umum	Jumlah perjanjian dengan penawaran umum sebagai penawaran visibilitas yang memiliki setidaknya 1 hari aktif dalam YTD.
Jumlah perjanjian penawaran pribadi	Jumlah perjanjian dengan penawaran pribadi sebagai penawaran visibilitas yang memiliki setidaknya 1 hari aktif dalam YTD.
Jumlah perjanjian penawaran umum baru	Jumlah perjanjian dengan penawaran umum sebagai visibilitas penawaran yang memiliki tanggal penerimaan dalam YTD.
Jumlah perjanjian penawaran pribadi baru	Jumlah perjanjian dengan penawaran pribadi sebagai visibilitas penawaran yang memiliki tanggal penerimaan dalam YTD.
Pelanggan aktif	Jumlah pelanggan unik, diidentifikasi sebagai ID AWS akun pelanggan, yang memiliki setidaknya 1 perjanjian aktif dalam YTD. Perjanjian aktif didefinisikan sebagai perjanjian dengan setidaknya 1 hari aktif selama periode waktu tersebut.
Pelanggan baru yang membayar	Jumlah pelanggan unik, diidentifikasi sebagai ID AWS akun pelanggan, yang memiliki bulan penagihan pertama mereka dalam YTD.

Bagian 3: Perjanjian penawaran publik dan pribadi

Bagian dasbor ini menampilkan ikhtisar perjanjian Anda. Indikator kinerja utama (KPIs) mencakup jumlah perjanjian penawaran umum, jumlah perjanjian penawaran pribadi, jumlah perjanjian

penawaran umum baru, dan jumlah perjanjian penawaran pribadi baru. Anda dapat melihat year-over-year atau period-over-period perubahan volume dan persentase. Anda dapat memperbarui rentang tanggal dengan memperbarui filter tanggal di bagian filter.

Perjanjian adalah kontrak yang ditandatangani antara pengusul (pemilik produk atau layanan) dan penerima (pelanggan) untuk mulai menggunakan produk.

 Note

Metrik perjanjian baru mencakup perjanjian aktif, kedaluwarsa, dibatalkan, dan diakhiri.

Tabel berikut mencantumkan dan menjelaskan status perjanjian.

Status	Deskripsi
Aktif	Beberapa atau semua ketentuan perjanjian berlaku.
Kedaluwarsa	Perjanjian berakhir pada tanggal akhir yang telah disepakati sebelumnya.
Dibatalkan	Akseptor memilih untuk mengakhiri perjanjian sebelum tanggal berakhirnya.
Diakhiri	Perjanjian berakhir sebelum tanggal akhir yang disepakati sebelumnya karena peristiwa penghentian yang diprakarsai AWS, seperti kegagalan pembayaran.
Diperbarui	Perjanjian itu diperbarui menjadi perjanjian baru menggunakan fungsionalitas seperti perpanjangan otomatis.
Diganti	Perjanjian itu diganti dengan tawaran pengganti.

Bagian 4: Metrik pelanggan

Bagian dasbor ini menyediakan tren pelanggan untuk rentang tanggal yang ditentukan. KPI termasuk jumlah pelanggan aktif dan jumlah pelanggan baru yang membayar.

- Tren bulanan — Memberikan tampilan bulanan untuk jumlah pelanggan aktif dan jumlah pelanggan baru yang membayar berdasarkan filter dalam bagan.
- Metrik pelanggan — Menyediakan data tentang jumlah pelanggan aktif dan jumlah pelanggan baru yang membayar, berdasarkan filter metrik yang dipilih. Anda dapat memilih segmen pelanggan atau industri untuk memahami bagaimana masing-masing berkontribusi pada metrik total pelanggan.

Note

Perjanjian memiliki status tertentu. Untuk informasi tentang status, lihat [tabel di bagian sebelumnya](#).

Tabel berikut mencantumkan dan menjelaskan metrik.

Metrik	Deskripsi
Perjanjian penawaran publik	Jumlah perjanjian yang diidentifikasi sebagai perjanjian IDs dengan visibilitas penawaran publik; panjang perjanjian memiliki setidaknya a satu hari tumpang tindih dengan rentang tanggal yang dipilih; panjangnya diidentifikasi antara tanggal mulai perjanjian dan tanggal akhir perjanjian.
Perjanjian penawaran pribadi	Jumlah perjanjian yang diidentifikasi sebagai perjanjian IDs dengan visibilitas penawaran pribadi; panjang perjanjian memiliki setidaknya a satu hari tumpang tindih dengan rentang tanggal yang dipilih; panjangnya diidentifikasi antara tanggal mulai perjanjian dan tanggal akhir perjanjian.

Metrik	Deskripsi
Perjanjian Penawaran Publik Baru	Jumlah perjanjian yang diidentifikasi sebagai ID perjanjian dengan visibilitas penawaran publik; tanggal penerimaan perjanjian termasuk dalam rentang tanggal yang dipilih.
Perjanjian penawaran pribadi baru	Jumlah perjanjian yang diidentifikasi sebagai perjanjian IDs dengan visibilitas penawaran pribadi; tanggal penerimaan perjanjian termasuk dalam rentang tanggal yang dipilih.
Pelanggan aktif	Jumlah pelanggan yang diidentifikasi sebagai AWS akun pelanggan IDs dengan setidaknya 1 perjanjian aktif; panjang perjanjian setidaknya satu hari tumpang tindih dengan rentang tanggal yang dipilih
Pelanggan baru yang membayar	Jumlah pelanggan yang diidentifikasi sebagai AWS akun pelanggan IDs yang memiliki bulan penagihan pertama mereka dalam rentang tanggal yang dipilih.

Untuk informasi selengkapnya tentang perjanjian dan pendapatan, lihat [Dasbor perjanjian dan pembaruan](#) dan [Dasbor pendapatan yang ditagih](#), keduanya di bagian ini.

Daftar dasbor kinerja

Dasbor kinerja daftar menyediakan ikhtisar, dan data terperinci tentang, daftar AWS Marketplace Anda. Dasbor menyediakan data tentang volume lalu lintas dan langkah-langkah yang diambil pelanggan Anda untuk berlangganan produk Anda. Dasbor juga menyediakan rincian rinci lalu lintas oleh saluran pemasaran.

Note

- Untuk membuka kunci dasbor ini, Anda harus mendaftarkan program [AWS Marketplace Penjual Prime](#).

- Untuk membuka dasbor ini, masuk ke [Portal Manajemen AWS Marketplace](#), pilih Wawasan, Pemasaran, lalu pilih tab Kinerja daftar.

Untuk informasi selengkapnya tentang menggunakan AWS Marketplace dasbor, lihat [Dasbor penjual](#), sebelumnya di bagian ini.

Topik

- [Bagian 1: Filter](#)
- [Bagian 2: Filter tanggal menyelam dalam](#)
- [Bagian 3: Ringkasan metrik](#)
- [Bagian 4: Tren bulanan dan konversi corong](#)
- [Bagian 5: Tren lalu lintas berdasarkan saluran](#)
- [Bagian 6: Metrik lalu lintas dan perjanjian bulanan](#)
- [Bagian 7: Sumber lalu lintas web untuk pengunjung unik](#)
- [Bagian 8: Kerusakan sumber lalu lintas web](#)

Bagian 1: Filter

Anda dapat menggunakan filter berikut untuk menyempurnakan data Anda.

Nama kontrol	Deskripsi
Judul produk	Judul produk.
Tanggal	Termasuk 30, 60, dan 90 hari terakhir, trailing 12 bulan (TTM), dan year to date (YTD). Anda dapat memilih kustom untuk menentukan tanggal mulai dan akhir tertentu.

Untuk informasi selengkapnya tentang pemfilteran, lihat [Memfilter data QuickSight di QuickSight Panduan Pengguna](#).

Bagian 2: Filter tanggal menyelam dalam

Filter tanggal ini berlaku untuk semua metrik pada dasbor kinerja daftar.

Misalnya, saat Anda menggunakan nilai year-to-date filter default, metrik berikut akan muncul:

Metrik	Deskripsi
Pengunjung Unqie	Jumlah individu unik yang telah mengunjungi halaman AWS Marketplace listing di YTD.
Perjanjian Penawaran Publik Baru	Jumlah perjanjian dengan penawaran umum sebagai penawaran visibilitas yang memiliki setidaknya 1 hari aktif dalam YTD.
Pengunjung unik baru	Jumlah individu unik yang telah mengunjungi halaman AWS Marketplace daftar untuk pertama kalinya di YTD.
Kembalikan pengunjung unik	Jumlah individu unik yang telah mengunjungi halaman AWS Marketplace listing yang sebelumnya telah mengunjungi sebelum awal tahun yang dipilih dan mengunjungi lagi di YTD.

Bagian 3: Ringkasan metrik

Bagian dasbor ini menampilkan visualisasi ringkasan dari lalu lintas dan perjanjian penawaran publik Anda. Indikator performance utama (KPIs) termasuk jumlah pengunjung unik, jumlah perjanjian penawaran umum baru, jumlah pengunjung unik baru, dan jumlah pengunjung unik yang kembali. Anda dapat melihat year-over-year atau period-over-period perubahan volume dan persentase. Anda dapat memperbarui rentang tanggal dengan memperbarui kriteria tanggal di bidang tanggal di bagian filter.

Metrik	Deskripsi
Pengunjung unik	Jumlah individu unik yang telah mengunjungi halaman AWS Marketplace daftar.
Perjanjian	Kontrak yang ditandatangani antara pengusul (pemilik produk atau layanan) dan penerima (pelanggan) untuk mulai menggunakan produk.

Metrik	Deskripsi
Perjanjian Penawaran Publik Baru	Jumlah perjanjian dengan penawaran umum sebagai visibilitas penawaran yang memiliki tanggal penerimaan dalam rentang tanggal yang dipilih. termasuk status aktif, kedaluwarsa, dibatalkan, dan dihentikan.
Status	Aktif - Beberapa atau semua ketentuan perjanjian berlaku. Kedaluwarsa - Perjanjian berakhir pada tanggal akhir yang telah disepakati sebelumnya. a. Dibatalkan - Akseptor memilih untuk mengakhiri perjanjian sebelum tanggal berakhirnya. Diakhiri - Perjanjian berakhir sebelum tanggal akhir yang disepakati sebelumnya karena peristiwa penghentian yang dimulai AWS, seperti kegagalan pembayaran. Diperpanjang - Perjanjian diperbarui menjadi perjanjian baru menggunakan fungsionalitas seperti perpanjangan otomatis. Diganti - Perjanjian diganti menggunakan penawaran pengganti.

Bagian 4: Tren bulanan dan konversi corong

Bagian dasbor ini menyediakan tren lalu lintas dan perjanjian untuk rentang tanggal tertentu. Indikator kinerja utama mencakup jumlah pengunjung unik dan jumlah perjanjian baru, mengacu pada perjanjian penawaran publik yang baru. Pengunjung unik berdasarkan bulan - memberikan tampilan bulanan untuk jumlah pengunjung unik. Total pengunjung unik bulanan bisa lebih dari total pengunjung unik di bagian metrik ringkasan karena satu pengunjung unik dapat mengunjungi beberapa bulan. Perjanjian baru berdasarkan bulan - memberikan tampilan bulanan untuk jumlah perjanjian penawaran publik baru. Konversi corong - menyediakan tampilan step-by-step konversi. Ini termasuk halaman daftar, halaman pengadaan, halaman konfigurasi dan halaman pemenuhan. Anda dapat memilih pengunjung unik, tampilan halaman, dan akun unik seperti yang ditampilkan oleh bar. Ketika Anda mengarahkan kursor ke setiap bar, Anda dapat melihat persentase konversi yang didefinisikan sebagai nilai bar saat ini sebagai persentase dari bar sebelumnya.

Metrik	Deskripsi
Pengunjung unik	Jumlah individu unik yang telah mengunjungi halaman AWS Marketplace daftar.
Tampilan halaman	Jumlah kunjungan ke halaman daftar AWS Marketplace.

Metrik	Deskripsi
Akun unik	Jumlah ID AWS akun unik yang terkait dengan pengunjung. Metrik ini hanya berlaku setelah pengunjung mengautentikasi.

Bagian 5: Tren lalu lintas berdasarkan saluran

Bagian dasbor ini memberikan penyelaman mendalam ke tren lalu lintas bulanan. Anda dapat memilih pengunjung unik dan tampilan halaman sebagai apa yang ditampilkan oleh bar. Anda dapat memilih satu saluran atau beberapa saluran di filter saluran untuk melihat tren bulanan untuk saluran yang dipilih.

Bagian 6: Metrik lalu lintas dan perjanjian bulanan

Bagian dasbor ini memberikan tampilan bulanan untuk pengunjung unik, perjanjian baru, tingkat konversi, dan total perjanjian. Tingkat konversi didefinisikan sebagai perjanjian baru yang dibagi oleh pengunjung unik untuk setiap bulan.

Data dalam tabel mewakili satu bulan penuh, bahkan jika Anda memilih rentang tanggal untuk sebagian bulan. Misalnya, jika Anda memfilter data untuk menampilkan periode dari 15 Januari 2024 hingga 15 Maret 2024, tabel tersebut masih akan menyertakan data bulanan lengkap untuk 1 Januari 2024 hingga 31 Maret 2024.

Data dalam tabel diurutkan berdasarkan Bulan, dengan bulan terakhir ditampilkan terlebih dahulu. Anda dapat memilih untuk mengurutkan tabel berdasarkan salah satu kolom yang tersedia dengan mengklik header kolom yang sesuai dengan opsi untuk mengurutkan data dalam urutan naik atau turun.

Bagian 7: Sumber lalu lintas web untuk pengunjung unik

Bagian dasbor ini memberikan penyelaman mendalam ke pengunjung unik oleh sumber pemasaran. Kategori termasuk penemuan organik penjual, kampanye pemasaran yang dipimpin penjual, penemuan AWS organik, dan kampanye yang dipimpin AWS. Kategori-kategori tersebut diidentifikasi oleh kode pelacakan yang terkait dengan pengunjung atau kunjungan.

Sumber lalu lintas	Deskripsi
Penemuan organik penjual	Pengunjung unik yang tiba di daftar Anda melalui ajakan bertindak dan titik penemuan di situs web Anda atau pemberitahuan dalam aplikasi.
Kampanye pemasaran yang dipimpin penjual	Pengunjung unik yang tiba di daftar Anda melalui iklan berbayar atau saluran promosi Anda.
AWS penemuan organik	Pengunjung unik yang tiba di daftar Anda melalui AWS saluran, seperti lalu lintas AWS situs web organik, pencarian AWS konsol, dan pencarian AWS Marketplace.
AWS kampanye pemasaran	Pengunjung unik yang tiba di daftar Anda melalui kampanye yang dipimpin AWS dan promosi berbayar.

Bagian 8: Kerusakan sumber lalu lintas web

Tabel rincian sumber lalu lintas web memberikan rincian granular kegiatan pemasaran yang mendorong lalu lintas.

- Sumber lalu lintas - Termasuk AWS-led dan seller-led sebagai dua kategori utama.
- Jenis lalu lintas - Termasuk penemuan organik penjual, kampanye pemasaran yang dipimpin penjual, penemuan AWS organik, dan kampanye pemasaran yang AWS dipimpin.
- Saluran - Termasuk berbagai taktik pemasaran, yang tampilan berbayar, pencarian berbayar, media sosial berbayar, email, organik, tautan eksternal lalu lintas situs internal dan lainnya.
- Penerbit - Mengacu pada entitas tempat kampanye diterbitkan, termasuk Facebook, Google,, LinkedIn AWS, Penjual-LED. dan lainnya.
- Promosi - Mengacu pada kampanye spesifik yang kode pelacakan khusus telah dibuat. Contohnya bisa jadipsm_linked_post_free-trial-global-ver-a.

Metrik	Deskripsi
Perjanjian yang dikaitkan	Total perjanjian yang disumbangkan oleh kunjungan untuk setiap promosi.
Tingkat konversi	Perjanjian yang dikaitkan dibagi oleh pengunjung unik untuk setiap promosi.

Dasbor kinerja pencarian

Dasbor kinerja Penelusuran menyediakan metrik tentang kinerja AWS Marketplace iklan Anda dalam pengalaman AWS Marketplace penelusuran. Anda menggunakan dasbor untuk meningkatkan kemampuan ditemukan dan kinerja daftar Anda. Dasbor melacak jumlah tayangan, klik, dan rasio klik tayang. Ini juga melacak kata kunci teratas yang Anda berikan, dan yang digunakan pelanggan, dan menawarkan saran kata kunci berdasarkan pencarian pelanggan umum.

Wawasan dari dasbor dapat membantu Anda menyempurnakan daftar produk Anda untuk meningkatkan visibilitas, mencocokkan kata kunci Anda dengan preferensi pelanggan, dan mengarahkan lalu lintas yang lebih berkualitas ke bisnis Anda. AWS Marketplace

Untuk membuka dasbor, mulai Portal AWS Marketplace Manajemen dan buka tab Wawasan.

Untuk informasi selengkapnya tentang mengontrol akses ke dasbor, lihat [Dasbor penjual](#), sebelumnya di bagian ini.

Bagian 1: Mengakses dasbor

Untuk membuka dasbor ini, Anda harus menjadi anggota [Program Prime AWS Marketplace Penjual](#).

Setelah Anda mendaftar di program Penjual Prime, administrator sistem untuk akun penjual dapat menggunakan tab Wawasan di Portal Manajemen AWS Marketplace untuk memulai dasbor kinerja Penelusuran.

Administrator sistem juga dapat membuat kebijakan AWS Identity and Access Management (IAM) untuk menyediakan akses dasbor tertentu ke pengguna lain di perusahaan penjual.

Note

Mulai September 2023, kami tidak lagi mendukung akses ke dasbor penjual yang diaktifkan oleh izin IAM lama. Gunakan format Amazon Resource Name (ARN) baru seperti yang ditunjukkan pada contoh berikut untuk memperbarui izin IAM Anda.

Untuk informasi selengkapnya tentang membuat kebijakan, lihat [Membuat kebijakan IAM](#), di Panduan AWS Identity and Access Management Pengguna.

Menggunakan kebijakan untuk mengontrol akses

Gunakan salah satu kebijakan berikut untuk menyediakan akses ke dasbor kinerja daftar.

Contoh berikut menyediakan akses ke semua AWS Marketplace sumber daya saat ini dan masa depan, termasuk dasbor dan laporan, terlepas dari umpan data saat ini atau masa depan.

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": [
      "aws-marketplace:GetSellerDashboard"
    ],
    "Resource": [
      "arn:aws:aws-marketplace::awsAccountID:AWSMarketplace/*",
    ]
  }]
}
```

Contoh berikut menyediakan akses ke dasbor kinerja Pencarian dengan memasukkan ARN-nya.

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Effect": "Allow",
    "Action": [
      "aws-marketplace:GetSellerDashboard"
    ],
    "Resource": [
      "arn:aws:aws-marketplace::awsAccountID:AWSMarketplace/ReportingData/Marketing_V1/
Dashboard/SearchPerformance_V1"
    ]
  }]
}
```

```
}]
}
```

Bagian 2: Filter

Bagian dasbor ini menyediakan filter berikut. Gunakan mereka untuk memperbaiki data Anda.

Nama kontrol	Deskripsi
Judul produk	<p>Judul produk.</p> <p>Filter ini memengaruhi tayangan, klik, rasio klik-tayang, dan rekomendasi kata kunci pencarian . Itu tidak mempengaruhi Top dicari kata kunci berdasarkan tabel kategori produk.</p>
Filter tanggal	<p>Anda dapat memfilter pada 30, 60, dan 90 hari terakhir, trailing 12 bulan (TTM), dan tahun ke tanggal (YTD). Anda dapat memilih Custom untuk menggunakan tanggal mulai dan berakhir tertentu.</p> <p>Filter ini memengaruhi tayangan, klik, dan rasio klik-tayang. Itu tidak memengaruhi rekomendasi kata kunci AWS Marketplace pencarian atau Kata kunci yang dicari teratas berdasarkan tabel kategori produk.</p>

Untuk informasi selengkapnya tentang pemfilteran, lihat [Memfilter data di Amazon QuickSight](#), di QuickSightPanduan Pengguna Amazon.

Bagian 3: kinerja AWS Marketplace pencarian

Bagian dasbor ini menampilkan indikator kinerja utama (KPIs) tentang kemampuan ditemukan dan kinerja listing Anda AWS Marketplace . Dasbor KPIs menyediakan jumlah tayangan, jumlah klik, dan rasio klik-tayang.

Anda dapat melihat year-over-year atau period-over-period mengubah volume dan persentase. Anda juga dapat mengubah rentang tanggal.

Metrik	Deskripsi
Tayangan	Berapa kali daftar Anda muncul di 20 hasil pencarian teratas di AWS Marketplace.
Klik	Berapa kali pengguna mengklik daftar Anda dari hasil AWS Marketplace pencarian.
Rasio klik-tayang (CTR)	Rasio antara jumlah klik dan jumlah tayangan. Ukuran seberapa efektif listing Anda dalam melibatkan pelanggan dari hasil pencarian.

Bagian 4: Tren bulanan untuk tayangan, klik, dan rasio klik-tayang

Bagian dasbor ini menyediakan tren bulanan untuk rentang tanggal tertentu. Ini KPIs menyediakan jumlah tayangan, jumlah klik, dan rasio klik-tayang.

Bagian 5: AWS Marketplace mencari rekomendasi kata kunci

Bagian dasbor ini menyediakan tabel yang mencantumkan kata kunci Anda saat ini, ditambah kata kunci yang direkomendasikan untuk setiap daftar Anda.

Nama kolom	Deskripsi
Judul produk	Judul produk. Tautan akan membawa Anda ke halaman ikhtisar produk di Portal AWS Marketplace Manajemen, tempat Anda dapat mengelola dan mengedit daftar Anda.
Kata kunci saat ini	Kata kunci yang Anda berikan di bagian Perbarui informasi produk di Portal AWS Marketplace Manajemen.
Kata kunci berkinerja terbaik	Kata kunci yang muncul dalam lima pencarian pelanggan teratas untuk produk Anda.
AWS-Kata kunci yang direkomendasikan	Kata kunci yang paling sering digunakan pelanggan saat mencari produk serupa.
Perkiraan keuntungan lalu lintas%	Potensi keuntungan lalu lintas jika Anda mengadopsi kata kunci yang direkomendasikan. Tidak terpengaruh oleh filter tanggal.

Note

Sementara kata kunci penting, faktor lain dapat mempengaruhi hasil pencarian.

- Pengoptimalan daftar
 - Optimalkan bagian lain dari daftar Anda, seperti deskripsi pendek dan panjang.
 - Pertimbangkan untuk menawarkan fitur seperti uji coba gratis untuk meningkatkan daya tarik iklan Anda.
- Mencocokkan kata kunci dengan daftar dan target pelanggan
 - Sebisa mungkin, cocokkan daftar Anda dengan kata kunci yang sering digunakan.
 - Gunakan hanya kata kunci yang relevan dengan produk Anda dan fitur-fiturnya. Jika tidak, daftar Anda mungkin muncul dalam jumlah pencarian yang lebih besar, tetapi lebih sedikit pengguna yang akan memilihnya.
- Memahami variasi kata kunci
 - Ingat bahwa istilah serupa, seperti “git” dan “gitops” dapat diperlakukan sebagai istilah pencarian yang sama.
 - Evaluasi relevansi variasi apa pun dan sesuaikan kata kunci Anda.
- Memperbarui kata kunci
 1. Di Portal Manajemen AWS Marketplace, buka menu Produk dan pilih kategori produk.
 2. Pilih tautan ke produk Anda, pilih tab Informasi produk, dan arahkan ke halaman Berikan informasi produk.
 3. Gulir ke bawah ke bagian AWS Marketplace kemampuan ditemukan, temukan Kata Kunci untuk hasil AWS Marketplace pencarian, dan sesuaikan kata kunci Anda.

Bagian 6: Kata kunci yang dicari teratas berdasarkan kategori produk

Lima kata kunci pencarian pelanggan teratas di setiap kategori produk untuk produk yang mirip dengan Anda. Filter produk dan tanggal lain di dasbor ini tidak memengaruhi data ini.

Memahami Beli dengan AWS dasbor

AWS Dasbor Beli dengan memberikan gambaran umum tentang lalu lintas web, keterlibatan, dan perjanjian yang dibuat oleh pelanggan yang memilih tombol Beli dengan AWS call-to-action tombol dan kunjungi halaman Beli dengan AWS pengadaan Anda di. AWS Marketplace

Untuk informasi tentang mengakses dasbor, lihat [Dasbor penjual](#), sebelumnya dalam panduan ini.

Topik

- [Bagian 1: Filter](#)
- [Bagian 2: Ringkasan metrik](#)
- [Bagian 3: Total klik dan data tren untuk tampilan halaman dan total perjanjian](#)
- [Bagian 4: AWS tingkat drop-off halaman masuk](#)
- [Bagian 5: Keterlibatan pada halaman Beli dengan AWS pengadaan](#)
- [Bagian 6: Data tren untuk metrik keterlibatan di halaman Beli dengan AWS pengadaan](#)

Bagian 1: Filter

Bagian ini menyediakan filter data berikut.

Filter nama	Deskripsi
Judul produk	Judul produk.
ID Produk	Pengidentifikasi unik yang ramah untuk produk perangkat lunak.
Filter tanggal	Filter menurut 30, 60, atau 90 hari terakhir, akhir 12 bulan, atau tahun hingga saat ini. Pilih Kustom untuk memasukkan tanggal mulai dan berakhir lainnya.

Untuk informasi selengkapnya tentang pemfilteran, lihat [Memfilter data QuickSight di QuickSight Panduan Pengguna](#).

Bagian 2: Ringkasan metrik

Bagian ini menampilkan informasi tentang AWS call-to-action tombol Beli dengan dan halaman Beli dengan AWS pengadaan.

Tabel berikut mencantumkan dan menjelaskan metrik. Untuk memperbarui rentang tanggal, ubah kriteria tanggal di bagian Filter.

Metrik	Deskripsi
Total klik tombol	Jumlah total klik pada call-to-action tombol, seperti Beli dengan AWS, di situs web penjual.
Jumlah Beli dengan tampilan AWS halaman	Jumlah total kunjungan ke halaman Beli dengan AWS pengadaan. Ini termasuk kunjungan berulang.
Total perjanjian dari Beli dengan AWS halaman	Jumlah total perjanjian yang dibuat di halaman Beli dengan AWS pengadaan.

Bagian 3: Total klik dan data tren untuk tampilan halaman dan total perjanjian

Bagian ini menyediakan tren bulanan atau mingguan untuk rentang tanggal tertentu. Tabel berikut mencantumkan dan menjelaskan KPIs:

Metrik	Deskripsi
Total klik tombol	Total klik pada call-to-action tombol di situs web penjual.
Tampilan halaman	Total kunjungan halaman ke halaman Buy with AWS procurement. Gunakan tampilan data menurut daftar untuk memilih antara tampilan bulanan dan mingguan.
Total perjanjian	Total perjanjian yang dibuat di halaman pengadaan Beli dengan AWS. Gunakan tampilan data menurut daftar untuk memilih antara tampilan bulanan dan mingguan.

Bagian 4: AWS tingkat drop-off halaman masuk

Bagian ini menampilkan tingkat drop-off untuk halaman AWS login. Tarifnya adalah rasio pengguna yang masuk dan membuat perjanjian dengan jumlah pengguna yang hanya masuk. Gunakan Tampilkan data menurut daftar untuk memfilter menurut minggu atau bulan.

Bagian 5: Keterlibatan pada halaman Beli dengan AWS pengadaan

Bagian ini menyediakan metrik ringkasan dan data tren untuk rentang tanggal tertentu. Tabel berikut mencantumkan dan menjelaskan metrik. Gunakan bidang tanggal di bagian filter untuk memperbarui rentang tanggal.

Metrik	Deskripsi
Jumlah Total Pengunjung Unik	Jumlah total pengguna halaman pengadaan unik.
Tingkat pentalan	Rasio pengguna yang masuk dan keluar setelah melihat satu halaman situs dengan mereka yang mengunjungi lebih dari satu halaman.
Rata-rata waktu tinggal pengguna dalam hitungan menit	Waktu antara saat pengguna masuk dan saat pengguna tersebut meninggalkan halaman.

Bagian 6: Data tren untuk metrik keterlibatan di halaman Beli dengan AWS pengadaan

Grafik menampilkan tren bulanan atau mingguan untuk metrik di [bagian 5](#). Gunakan tampilan data menurut daftar untuk memilih antara tampilan bulanan dan mingguan.

AWS Marketplace Wawasan Vendor

AWS Marketplace Vendor Insights adalah fitur yang menyederhanakan penilaian risiko perangkat lunak yang dilakukan oleh organisasi untuk melindungi pengadaan perangkat lunak yang mereka percayai dan memenuhi standar mereka. Dengan AWS Marketplace Vendor Insights, pembeli dapat memantau profil keamanan produk dalam waktu dekat dari satu konsol. AWS Marketplace Vendor Insights dapat memudahkan proses pengadaan bagi pembeli dan berpotensi meningkatkan penjualan bagi penjual. Ini mengurangi upaya penilaian pembeli dengan menyediakan dasbor informasi keamanan dan kepatuhan produk perangkat lunak.

Semua informasi keamanan dan kepatuhan di dasbor Wawasan AWS Marketplace Vendor didasarkan pada bukti yang dikumpulkan dari sumber berikut:

- Pengesahan diri Penjual, termasuk penilaian mandiri keamanan AWS Marketplace Vendor Insights dan Consensus Assessment Initiative Questionnaire (CAIQ)
- Laporan audit standar industri (misalnya, Organisasi Internasional untuk Standardisasi ISO 27001)
- AWS Audit Manager, yang mengotomatiskan pengumpulan bukti dari lingkungan produksi penjual

AWS Marketplace Vendor Insights mengumpulkan artefak kepatuhan dan informasi kontrol keamanan tentang produk dan menyajikannya di dasbor. Dasbor mengambil data dari penilaian diri penjual, bukti dari laporan audit, dan bukti langsung dari Akun AWS. Data ini dimasukkan ke dalam kontrol keamanan dan kemudian ke dasbor untuk ditinjau pembeli. Bukti langsung adalah metode memperbarui data secara konsisten dari berbagai sumber untuk menyajikan informasi terkini. AWS Config diaktifkan di lingkungan penjual. Data tentang konfigurasi, backup diaktifkan, dan informasi lainnya diperbarui secara otomatis. Misalnya, asumsikan bahwa Kontrol Akses untuk suatu produk sesuai dan bucket Amazon Simple Storage Service (Amazon S3) menjadi publik. Dasbor akan menampilkan bahwa status kontrol berubah dari Compliant ke Undetermined.

Anda harus menyiapkan sumber daya dan infrastruktur dasar Akun AWS sebelum menggunakan Wawasan AWS Marketplace Vendor. Setelah penyiapan selesai, AWS Marketplace Vendor Insights dapat mengumpulkan informasi dan menghasilkan profil keamanan untuk produk perangkat lunak Anda sebagai layanan (SaaS) di AWS Marketplace

Konten

- [Memahami Wawasan AWS Marketplace Vendor](#)
- [Menyiapkan Wawasan AWS Marketplace Vendor](#)

- [Melihat profil AWS Marketplace Vendor Insights Anda](#)
- [Mengelola snapshot di Wawasan AWS Marketplace Vendor](#)
- [Mengontrol akses di Wawasan AWS Marketplace Vendor](#)

Memahami Wawasan AWS Marketplace Vendor

AWS Marketplace Vendor Insights mengumpulkan artefak kepatuhan dan informasi kontrol keamanan untuk produk Anda dan menyajikannya di dasbor. Dasbor mengambil data dari penilaian diri pemilik produk, bukti dari laporan audit, dan bukti langsung dari Akun AWS. Data ini dimasukkan ke dalam kontrol keamanan dan kemudian ke dasbor untuk ditinjau pembeli.

Dasbor menyajikan informasi berbasis bukti yang dikumpulkan oleh AWS Marketplace Vendor Insights dari beberapa kategori kontrol keamanan. Ini memberikan wawasan dengan tampilan dekat real-time dari profil keamanan dan mengurangi diskusi antara pembeli dan penjual. Pembeli dapat memvalidasi informasi penjual menyelesaikan penilaian dalam beberapa jam. AWS Marketplace Vendor Insights menyediakan mekanisme bagi penjual untuk menjaga keamanan dan informasi postur kepatuhan up-to-date secara otomatis. Mereka dapat membagikannya dengan pembeli sesuai permintaan yang menghilangkan kebutuhan untuk menanggapi kuesioner secara acak.

AWS Marketplace Vendor Insights mengumpulkan informasi berbasis bukti dari tiga sumber:

- Penilaian mandiri vendor Anda — Penilaian mandiri yang didukung mencakup penilaian mandiri keamanan AWS Marketplace Vendor Insights dan Consensus Assessment Initiative Questionnaire (CAIQ).
- Akun produksi Anda — Dari beberapa kontrol, 25 kontrol mendukung pengumpulan bukti langsung dari akun produksi Anda. Bukti langsung untuk setiap kontrol dihasilkan dengan mengevaluasi pengaturan konfigurasi AWS sumber daya Anda menggunakan satu atau beberapa AWS Config aturan. AWS Audit Manager menangkap bukti dan mempersiapkannya untuk dikonsumsi oleh AWS Marketplace Vendor Insights. AWS CloudFormation Templat orientasi mengotomatiskan langkah-langkah prasyarat yang diperlukan untuk memungkinkan pengumpulan bukti langsung. AWS Config diaktifkan di lingkungan penjual. Data tentang konfigurasi, backup diaktifkan, dan informasi lainnya diperbarui secara otomatis. Misalnya, asumsikan bahwa Kontrol Akses untuk suatu produk adalah Compliant dan bucket Amazon S3 menjadi publik. Dasbor akan menampilkan bahwa status kontrol berubah dari Compliant ke Undetermined.
 - Menghidupkan AWS Config dan AWS Audit Manager layanan.
 - Membuat AWS Config aturan dan penilaian AWS Audit Manager otomatis.

- Menyediakan peran AWS Identity and Access Management (IAM) sehingga AWS Marketplace Vendor Insights dapat menarik hasil penilaian.
- Laporan ISO 27001 dan SOC2 Tipe II Anda — Kategori kontrol dipetakan ke kontrol dalam laporan Organisasi Internasional untuk Standardisasi (ISO) dan Sistem dan Kontrol Organisasi (). SOC2 Saat Anda membagikan laporan ini dengan AWS Marketplace Vendor Insights, laporan tersebut dapat mengekstrak bukti yang relevan dari laporan ini dan menyajikannya di dasbor.

Menyiapkan Wawasan AWS Marketplace Vendor

Prosedur berikut menjelaskan langkah-langkah tingkat tinggi untuk menyiapkan Wawasan AWS Marketplace Vendor pada daftar AWS Marketplace perangkat lunak Anda sebagai layanan (SaaS).

Untuk mengatur Wawasan AWS Marketplace Vendor di daftar SaaS Anda

1. [the section called “Buat profil keamanan”](#).
2. (Opsional)[the section called “Unggah sertifikasi”](#).
3. [the section called “Unggah penilaian diri”](#).
4. (Opsional)[the section called “Aktifkan penilaian AWS Audit Manager otomatis”](#).

Buat profil keamanan

Profil keamanan memberi pembeli Anda wawasan terperinci tentang postur keamanan produk perangkat lunak Anda. Profil keamanan menggunakan sumber data terkait, termasuk penilaian mandiri, sertifikasi, dan AWS Audit Manager penilaian otomatis.

Note

Anda dapat membuat profil keamanan dalam jumlah terbatas. Untuk membuat lebih banyak profil keamanan, mintalah peningkatan kuota. Untuk informasi selengkapnya, lihat [kuota AWS layanan](#) di Referensi Umum AWS

Untuk membuat profil keamanan

1. Masuk menggunakan pengguna atau peran IAM dengan akses ke akun AWS Marketplace penjual.

2. Pilih Produk dan pilih SaaS untuk menavigasi ke halaman produk SaaS.
3. Pilih produk.
4. Pilih tab Vendor Insights, lalu pilih Contact Support untuk menambahkan profil keamanan.
5. Lengkapi formulir, lalu pilih Kirim.

Tim Operasi AWS Marketplace Penjual akan membuat profil keamanan. Ketika profil keamanan siap, mereka akan mengirim pesan email pemberitahuan ke penerima yang diidentifikasi pada formulir.

Unggah sertifikasi

Sertifikasi adalah sumber data yang memberikan bukti postur keamanan produk Anda di berbagai dimensi. AWS Marketplace Vendor Insights mendukung sertifikasi berikut:

- Sertifikasi FedRAMP - Memvalidasi kepatuhan dengan standar keamanan cloud pemerintah AS
- Laporan kepatuhan GDPR - Menunjukkan kepatuhan terhadap persyaratan Peraturan Perlindungan Data Umum (GDPR), melindungi data pribadi dan hak individu atas privasi
- Laporan kepatuhan HIPAA - Menunjukkan kepatuhan terhadap peraturan Undang-Undang Portabilitas dan Akuntabilitas Asuransi Kesehatan (HIPAA), menjaga informasi kesehatan yang dilindungi
- ISO/IEC 27001 audit report – Confirms compliance with International Organization for Standardization (ISO)/International Komisi Elektroteknik (IEC) 27001, menekankan standar keamanan informasi
- Laporan audit PCI DSS - Menunjukkan kepatuhan terhadap standar keamanan yang ditetapkan oleh Dewan Standar Keamanan PCI
- Laporan audit SOC 2 Tipe 2 — Mengonfirmasi kepatuhan terhadap kontrol privasi dan keamanan data Service Organizational Control (SOC)

Untuk mengunggah sertifikasi

1. Pada tab Wawasan Vendor, arahkan ke bagian Sumber data.
2. Di bawah Sertifikasi, pilih Unggah sertifikasi.
3. Di bawah rincian Sertifikasi, berikan informasi yang diminta dan unggah sertifikasi.
4. (Opsional) Di bawah Tag, tambahkan tag baru.

 Note

Untuk informasi tentang tag, lihat [Menandai AWS sumber daya Anda](#) di Panduan Pengguna AWS Sumber Daya Penandaan.

5. Pilih Unggah sertifikasi.

 Note

Sertifikasi secara otomatis dikaitkan dengan profil keamanan saat ini. Anda juga dapat mengaitkan sertifikasi yang sudah Anda unggah. Pada halaman detail produk, pilih Sertifikasi asosiasi di bawah Sertifikasi, pilih sertifikasi dari daftar, dan pilih Sertifikasi asosiasi.

Setelah Anda mengunggah sertifikasi, Anda dapat mengunduhnya menggunakan tombol Unduh sertifikasi di halaman detail produk. Anda juga dapat memperbarui detail sertifikasi menggunakan tombol Perbarui sertifikasi.

Status sertifikasi berubah ValidationPending hingga detail sertifikasi divalidasi. Status alternatif muncul selama dan setelah sumber data diproses:

- Tersedia — Sumber data diunggah dan validasi sistem berhasil diselesaikan.
- AccessDenied— Referensi sumber eksternal sumber data tidak lagi dapat diakses oleh AWS Marketplace Vendor Insights untuk dibaca.
- ResourceNotFoundReferensi sumber eksternal sumber data tidak lagi tersedia VendorInsights untuk dibaca.
- ResourceNotSupported— Sumber data telah diunggah tetapi sumber yang disediakan belum didukung. Untuk detail tentang kesalahan validasi, lihat pesan status.
- ValidationPending— Sumber data diunggah tetapi validasi sistem masih berjalan. Tidak ada item tindakan untuk Anda pada tahap ini. Status diperbarui ke Tersedia, ResourceNotSupported, atau ValidationFailed.
- ValidationFailedSumber data diunggah, tetapi validasi sistem gagal karena satu atau lebih alasan. Untuk detail tentang kesalahan validasi, lihat pesan status.

Unggah penilaian diri

Penilaian diri adalah jenis sumber data yang memberikan bukti postur keamanan produk Anda. AWS Marketplace Vendor Insights mendukung penilaian mandiri berikut:

- AWS Marketplace Penilaian mandiri Wawasan Vendor
- Kuesioner Inisiatif Penilaian Konsensus (CAIQ). Untuk informasi selengkapnya, lihat [Apa itu CAIQ](#), di situs web Cloud Security Alliance.

Untuk mengunggah penilaian diri

1. Buka AWS Marketplace konsol di <https://console.aws.amazon.com/marketplace>.
2. Pada tab Wawasan Vendor, arahkan ke bagian Sumber data.
3. Di bawah Penilaian diri, pilih Unggah penilaian diri.
4. Di bawah rincian penilaian diri, lengkapi informasi berikut:
 - a. Nama — Masukkan nama untuk penilaian diri.
 - b. Jenis - Pilih jenis penilaian dari daftar.

Note

Jika Anda memilih Penilaian Mandiri Keamanan Wawasan Vendor, pilih Unduh templat untuk mengunduh penilaian mandiri. Pilih Ya, Tidak, atau N/A untuk setiap jawaban di spreadsheet.

5. Untuk mengunggah penilaian yang telah selesai, pilih Unggah penilaian mandiri.
6. (Opsional) Di bawah Tag, tambahkan tag baru.

Note

Untuk informasi tentang tag, lihat [Menandai AWS sumber daya Anda](#) di Panduan Pengguna AWS Sumber Daya Penandaan.

7. Pilih Unggah penilaian diri.

Note

Penilaian diri secara otomatis dikaitkan dengan profil keamanan saat ini. Anda juga dapat mengaitkan penilaian diri yang sudah Anda unggah. Pada halaman detail produk, pilih Penilaian mandiri asosiasi di bawah Penilaian diri, pilih penilaian mandiri dari daftar, dan pilih Penilaian mandiri asosiasi.

Setelah Anda mengunggah penilaian diri, Anda dapat mengunduhnya menggunakan tombol Unduh penilaian mandiri di halaman detail produk. Anda juga dapat memperbarui detail penilaian diri menggunakan tombol Perbarui penilaian diri.

Status diperbarui ke salah satu dari berikut ini:

- **Tersedia** — Sumber data diunggah dan validasi sistem berhasil diselesaikan.
- **AccessDeniedReferensi** sumber eksternal sumber data tidak lagi tersedia VendorInsights untuk dibaca.
- **ResourceNotFoundReferensi** sumber eksternal sumber data tidak lagi tersedia VendorInsights untuk dibaca.
- **ResourceNotSupported**— Sumber data telah diunggah tetapi sumber yang disediakan belum didukung. Untuk detail tentang kesalahan validasi, lihat pesan status.
- **ValidationPending**— Sumber data diunggah, tetapi validasi sistem masih berjalan. Tidak ada item tindakan untuk Anda pada tahap ini. Status diperbarui ke Tersedia, ResourceNotSupported, atau ValidationFailed.
- **ValidationFailed** Sumber data diunggah, tetapi validasi sistem gagal karena satu atau lebih alasan. Untuk detail tentang kesalahan validasi, lihat pesan status.

Aktifkan penilaian AWS Audit Manager otomatis

AWS Marketplace Vendor Insights menggunakan beberapa Layanan AWS untuk secara otomatis mengumpulkan bukti untuk profil keamanan Anda.

Anda memerlukan yang berikut Layanan AWS dan sumber daya untuk penilaian otomatis:

- **AWS Audit Manager**— Untuk menyederhanakan penyiapan AWS Marketplace Vendor Insights, kami menggunakan AWS CloudFormation Stacks dan StackSets, yang menangani penyediaan dan

konfigurasi sumber daya yang diperlukan. Kumpulan tumpukan membuat penilaian otomatis yang berisi kontrol yang diisi secara otomatis oleh AWS Config.

Untuk informasi selengkapnya AWS Audit Manager, lihat [Panduan AWS Audit Manager Pengguna](#).

- AWS Config— Set tumpukan menyebarkan paket AWS Config kesesuaian untuk mengatur aturan yang diperlukan. AWS Config Aturan ini memungkinkan penilaian otomatis Audit Manager untuk mengumpulkan bukti langsung untuk orang lain yang Layanan AWS digunakan dalam hal itu Akun AWS. Untuk informasi selengkapnya tentang AWS Config fitur, lihat [Panduan AWS Config Pengembang](#).

Note

Anda mungkin melihat peningkatan aktivitas di akun Anda selama bulan awal perekaman AWS Config dibandingkan dengan bulan-bulan berikutnya. Selama proses bootstrap awal, tinjau semua AWS Config sumber daya di akun Anda yang telah Anda pilih untuk AWS Config direkam.

Jika Anda menjalankan beban kerja sementara, Anda mungkin melihat peningkatan aktivitas dari AWS Config saat merekam perubahan konfigurasi yang terkait dengan pembuatan dan penghapusan sumber daya sementara ini. Beban kerja sementara adalah penggunaan sementara sumber daya komputasi yang dimuat dan dijalankan saat diperlukan.

Contoh beban kerja sementara termasuk instans spot Amazon Elastic Compute Cloud (Amazon EC2), pekerjaan EMR Amazon, dan. AWS Auto Scaling AWS Lambda Untuk menghindari peningkatan aktivitas menjalankan beban kerja sementara, Anda dapat menjalankan jenis beban kerja ini di akun terpisah dengan dimatikan. AWS Config Pendekatan ini menghindari peningkatan perekaman konfigurasi dan evaluasi aturan.

- Amazon S3 - Kumpulan tumpukan membuat dua bucket Amazon Simple Storage Service berikut:
 - vendor-insights-stack-set-output-bucket- {account number} — Bucket ini berisi output dari stack set run. Tim Operasi AWS Marketplace Penjual menggunakan output untuk menyelesaikan proses pembuatan sumber data otomatis Anda.
 - vendor-insights-assessment-reports-bucket- {nomor akun} - AWS Audit Manager menerbitkan laporan penilaian ke bucket Amazon S3 ini. Untuk informasi selengkapnya tentang menerbitkan laporan [penilaian](#), lihat [Laporan penilaian](#) di Panduan AWS Audit Manager Pengguna.

Untuk informasi selengkapnya tentang fitur Amazon S3, lihat Panduan Pengguna [Amazon S3](#).

- IAM - Kumpulan orientasi menetapkan peran berikut AWS Identity and Access Management (IAM) di akun Anda:
 - Ketika `VendorInsightsPrerequisiteCFT.yml` template digunakan, itu menciptakan peran administrator `AWSVendorInsightsOnboardingStackSetsAdmin` dan peran `AWSVendorInsightsOnboardingStackSetsExecution` run. Kumpulan tumpukan menggunakan peran administrator untuk menyebarkan tumpukan yang diperlukan ke beberapa Wilayah AWS secara bersamaan. Peran administrator mengasumsikan peran eksekusi untuk menyebarkan induk dan tumpukan bersarang yang diperlukan sebagai bagian dari proses persiapan Wawasan AWS Marketplace Vendor. Untuk informasi selengkapnya tentang izin yang dikelola sendiri, lihat [Memberikan izin yang dikelola sendiri di Panduan Pengguna](#).AWS CloudFormation
 - `AWSVendorInsightsRole` Peran ini memberi AWS Marketplace Vendor Insights akses untuk membaca penilaian dalam AWS Audit Manager sumber daya. AWS Marketplace Wawasan Vendor menampilkan bukti yang ditemukan pada penilaian di profil Wawasan AWS Marketplace Vendor Anda.
 - AWS Marketplace Vendor Insights `AWSVendorInsightsOnboardingDelegationRole` menyediakan akses ke daftar dan membaca objek di `vendor-insights-stack-set-output-bucket` bucket. Kemampuan ini memungkinkan tim Operasi AWS Marketplace Katalog untuk membantu Anda menyiapkan profil Wawasan AWS Marketplace Vendor.
 - `AWSAuditManagerAdministratorAccess` Peran ini menyediakan akses administratif untuk mengaktifkan atau menonaktifkan AWS Audit Manager, memperbarui pengaturan, dan mengelola penilaian, kontrol, dan kerangka kerja. Anda atau tim Anda dapat mengambil peran ini untuk mengambil tindakan untuk penilaian otomatis di AWS Audit Manager.

Untuk mengaktifkan penilaian AWS Audit Manager otomatis, Anda harus menerapkan tumpukan orientasi.

Terapkan tumpukan orientasi

Untuk menyederhanakan persiapan AWS Marketplace Vendor Insights, kami menggunakan AWS CloudFormation Stacks dan StackSets, yang menangani penyediaan dan konfigurasi sumber daya yang diperlukan. Jika Anda memiliki beberapa akun atau beberapa solusi Wilayah AWS SaaS, StackSets izinkan Anda menerapkan tumpukan orientasi dari akun manajemen pusat.

Untuk informasi selengkapnya CloudFormation StackSets, lihat [Bekerja dengan AWS CloudFormation StackSets](#) di Panduan AWS CloudFormation Pengguna.

AWS Marketplace Penyiapan Vendor Insights mengharuskan Anda menggunakan CloudFormation templat berikut:

- `VendorInsightsPrerequisiteCFT`— Mengatur peran dan izin administrator yang diperlukan untuk dijalankan CloudFormation StackSets di akun Anda. Buat tumpukan ini di akun penjual Anda.
- `VendorInsightsOnboardingCFT`— Mengatur yang diperlukan Layanan AWS dan mengonfigurasi izin IAM yang sesuai. Izin ini memungkinkan Wawasan AWS Marketplace Vendor mengumpulkan data untuk produk SaaS yang berjalan di Anda Akun AWS dan menampilkan data di profil Wawasan Vendor Anda AWS Marketplace . Buat tumpukan ini di akun penjual dan akun produksi Anda yang menghosting solusi SaaS Anda. StackSets

Buat `VendorInsightsPrerequisiteCFT` tumpukan

Dengan menjalankan `VendorInsightsPrerequisiteCFT` CloudFormation tumpukan, Anda mengatur izin IAM untuk memulai kumpulan tumpukan orientasi.

Untuk membuat `VendorInsightsPrerequisiteCFT` tumpukan

1. Tinjau dan unduh `VendorInsightsPrerequisiteCFT.yml` file terbaru dari [folder template AWS Sampel Repo for Vendor Insights di situs](#) web. GitHub
2. Masuk ke akun AWS Management Console menggunakan AWS Marketplace penjual Anda, lalu buka AWS CloudFormation konsol di <https://console.aws.amazon.com/cloudformation>.
3. Di panel navigasi CloudFormation konsol, pilih Tumpukan, lalu pilih Buat tumpukan dan Dengan sumber daya baru (standar) dari dropdown. (Jika panel navigasi tidak terlihat, di sudut kiri atas, pilih dan perluas panel navigasi.)
4. Di bawah Tentukan templat, pilih Unggah file templat. Untuk mengunggah `VendorInsightsPrerequisiteCFT.yml` file yang Anda unduh, gunakan Pilih file. Lalu pilih Selanjutnya.
5. Masukkan nama untuk tumpukan, lalu pilih Berikutnya.
6. (Opsional) Konfigurasi opsi tumpukan yang Anda inginkan.

Pilih Berikutnya.

7. Pada halaman Tinjau, tinjau pilihan Anda. Untuk membuat perubahan, pilih Edit di area yang ingin Anda ubah. Sebelum Anda dapat membuat tumpukan, Anda harus memilih kotak centang pengakuan di area Kemampuan.

Pilih Kirim.

8. Setelah tumpukan dibuat, pilih tab Resources dan catat peran berikut yang dibuat:

- `AWSVendorInsightsOnboardingStackSetsAdmin`
- `AWSVendorInsightsOnboardingStackSetsExecution`

Buat `VendorInsightsOnboardingCFT` set tumpukan

Dengan menjalankan set `VendorInsightsOnboardingCFT` CloudFormation tumpukan, Anda mengatur yang diperlukan Layanan AWS dan mengonfigurasi izin IAM yang sesuai. Hal ini memungkinkan Wawasan AWS Marketplace Vendor mengumpulkan data untuk produk SaaS yang berjalan di Akun AWS Anda dan menampilkannya di profil Wawasan Vendor AWS Marketplace Anda.

Jika Anda memiliki beberapa solusi akun atau jika Anda memiliki akun penjual dan produksi terpisah, Anda harus menerapkan tumpukan ini di beberapa akun. StackSets memungkinkan Anda melakukan ini dari akun manajemen tempat Anda membuat tumpukan prasyarat.

Kumpulan tumpukan digunakan menggunakan izin yang dikelola sendiri. Untuk informasi lebih lanjut, lihat [Membuat set tumpukan dengan izin dikelola sendiri](#) di Panduan Pengguna AWS CloudFormation

Untuk membuat `VendorInsightsOnboardingCFT` set tumpukan

1. Tinjau dan unduh `VendorInsightsOnboardingCFT.yml` file terbaru dari [folder template AWS Sampel Repo for Vendor Insights di situs](#) web. GitHub
2. Masuk ke akun AWS Management Console menggunakan AWS Marketplace penjual Anda, lalu buka AWS CloudFormation konsol di <https://console.aws.amazon.com/cloudformation..>
3. Di panel navigasi CloudFormation konsol, pilih Buat StackSet. (Jika panel navigasi tidak terlihat, di sudut kiri atas, pilih dan perluas panel navigasi.)
4. Di bawah Izin, untuk peran administrator pilih nama peran IAM, lalu pilih `AWSVendorInsightsOnboardingStackSetsAdmin` nama peran dari menu tarik-turun.
5. Masukkan **`AWSVendorInsightsOnboardingStackSetsExecution`** sebagai nama peran eksekusi IAM.
6. Di bawah Tentukan templat, pilih Unggah file templat. Untuk mengunggah `VendorInsightsOnboardingCFT.yml` file yang Anda unduh, gunakan Pilih file lalu pilih Berikutnya.
7. Berikan StackSet parameter berikut, lalu pilih Berikutnya.

- `CreateVendorInsightsAutomatedAssessment`— Parameter ini mengatur penilaian AWS Audit Manager otomatis di Anda Akun AWS. Jika Anda memiliki akun manajemen dan produksi terpisah, opsi ini hanya boleh dipilih untuk akun produksi dan bukan untuk akun manajemen.
 - `CreateVendorInsightsIAMRoles`— Parameter ini menyediakan peran IAM yang memungkinkan AWS Marketplace Vendor Insights untuk membaca data penilaian di Anda. Akun AWS
 - `PrimaryRegion`— Parameter ini menetapkan primer Wilayah AWS untuk penyebaran SaaS Anda. Ini adalah Wilayah tempat bucket Amazon S3 dibuat di Anda. Akun AWS Jika produk SaaS Anda dikerahkan hanya ke satu Wilayah, Wilayah tersebut adalah Wilayah utama.
8. Konfigurasi StackSet opsi yang Anda inginkan. Pertahankan konfigurasi Eksekusi sebagai Tidak Aktif, lalu pilih Berikutnya.
 9. Konfigurasi opsi penerapan. Jika Anda memiliki solusi beberapa akun, Anda dapat mengonfigurasi kumpulan tumpukan untuk diterapkan di beberapa akun dan Wilayah sebagai satu operasi. Pilih Berikutnya.

Note

Jika Anda memiliki solusi beberapa akun, kami tidak menyarankan untuk menerapkan ke semua akun sebagai satu set tumpukan. Perhatikan baik-baik parameter yang ditentukan pada langkah 7. Anda mungkin ingin mengaktifkan atau menonaktifkan beberapa parameter, tergantung pada jenis akun yang Anda gunakan. StackSets terapkan parameter yang sama ke semua akun yang ditentukan dalam satu penerapan. Anda dapat mengurangi waktu penerapan dengan mengelompokkan akun dalam kumpulan tumpukan, tetapi Anda masih perlu menerapkan beberapa kali untuk solusi beberapa akun.

Important

Jika Anda menerapkan ke beberapa Wilayah, Wilayah pertama yang Anda cantumkan haruslah. `PrimaryRegion` Biarkan opsi `Region Concurrency` sebagai pengaturan default `Sequential`.

10. Pada halaman Tinjau, tinjau pilihan Anda. Untuk membuat perubahan, pilih Edit di area yang ingin Anda ubah. Sebelum Anda dapat membuat kumpulan tumpukan, Anda harus memilih kotak centang pengakuan di area Capabilities.

Pilih Kirim.

Set tumpukan membutuhkan waktu sekitar 5 menit per Wilayah untuk diselesaikan.

Melihat profil AWS Marketplace Vendor Insights Anda

Profil Anda di AWS Marketplace Vendor Insights memberikan informasi penting bagi pembeli untuk digunakan saat mereka menilai produk Anda. Untuk tujuan perlindungan data, kami menyarankan Anda untuk melindungi Akun AWS kredensial Anda dan mengatur pengguna individu dengan AWS Identity and Access Management (IAM). Dengan pendekatan itu, setiap pengguna hanya diberikan izin yang diperlukan untuk memenuhi tugas pekerjaan mereka. Untuk informasi selengkapnya tentang membuat pengguna dengan IAM, lihat [the section called “Membuat atau menggunakan grup”](#).

Pembeli dapat menilai produk Anda dengan menggunakan dasbor AWS Marketplace Vendor Insights. Di sana, pembeli dapat melihat ikhtisar produk yang ditentukan oleh sumber data yang Anda tambahkan ke profil Anda. Profil keamanan ditentukan oleh beberapa kontrol keamanan dalam 10 kategori.

10 kategori keamanan yang digunakan untuk mendefinisikan data adalah sebagai berikut:

- Manajemen akses
- Keamanan aplikasi
- Audit, kepatuhan, dan kebijakan keamanan
- Ketahanan bisnis
- Keamanan dan privasi data
- Perangkat pengguna akhir dan keamanan seluler
- Sumber daya manusia
- Keamanan infrastruktur
- Manajemen risiko dan respons insiden
- Kebijakan keamanan dan konfigurasi

Untuk informasi selengkapnya, lihat [Memahami kategori kontrol](#), di Panduan AWS Marketplace Pembeli.

Dengan menyiapkan dan menggunakan Wawasan AWS Marketplace Penjual, Anda setuju untuk mematuhi Layanan AWS persyaratan dan aturan privasi data untuk menjaga informasi pengguna tetap pribadi dan aman. Untuk informasi selengkapnya tentang persyaratan privasi AWS data, lihat [FAQ Privasi Data](#). Untuk informasi selengkapnya tentang persyaratan layanan, lihat [Layanan AWS persyaratan](#).

Lihat profil keamanan Anda sebagai penjual

Setelah menyelesaikan penilaian diri dan menambahkan bukti langsung lainnya, penting untuk melihat profil Anda sebagai penjual. Anda akan ingin meninjau informasi yang ditambahkan ke profil Anda.

Note

Profil ini tidak dapat dilihat oleh pembeli hingga Anda meminta tim dukungan AWS Marketplace Vendor Insights memperbarui visibilitasnya. Setelah tim dukungan menyelesaikan pembaruan, profil keamanan dapat diakses oleh pembeli yang menandatangani perjanjian kerahasiaan (NDA) Anda.

Jika Anda ingin menghapus data informasi identitas pribadi (PII) pelanggan dari profil AWS Marketplace Vendor Insights Anda, mulailah kasus dukungan dengan menghubungi

[Dukungan](#)

Untuk melihat profil keamanan Anda saat pembeli melihatnya

1. Masuk ke AWS Management Console.
2. Buka halaman [Produk SaaS](#) di portal.
3. Pilih produk dengan profil keamanan terkait.
4. Pilih tab Wawasan Vendor, lalu pilih Lihat Snapshot Rilis Terbaru.
5. Pada tab Ikhtisar, semua rencana sertifikat yang Anda unggah ditampilkan.
6. Pilih tab Keamanan dan kepatuhan, tempat Anda dapat melihat data yang dikumpulkan dari beberapa kontrol. Untuk melihat detail selengkapnya, pilih setiap set kontrol.

Mengelola snapshot di Wawasan AWS Marketplace Vendor

Snapshot adalah point-in-time postur profil keamanan. Di AWS Marketplace Vendor Insights, Anda dapat menggunakan snapshot untuk menilai produk penjual pada waktu tertentu. Sebagai penjual, Anda dapat membandingkan postur keamanan profil Anda pada waktu yang berbeda atau snapshot terbaru dari profil keamanan yang berbeda untuk mendukung pengambilan keputusan Anda. Snapshot memberikan informasi keamanan yang diperlukan selain memberikan transparansi tentang kesegaran dan sumber data.

Di AWS Marketplace konsol, di bagian ringkasan Snapshot Wawasan AWS Marketplace Vendor, Anda dapat melihat detail snapshot berikut untuk jadwal pembuatan dan rilis:

- Snapshot terakhir dibuat — Snapshot terakhir dibuat untuk profil ini.
- Pembuatan terjadwal berikutnya - Snapshot dijadwalkan akan dibuat berikutnya.
- Frekuensi pembuatan - Lama waktu antara kreasi snapshot atau frekuensi pembuatan snapshot.
- Rilis terjadwal berikutnya — Snapshot dijadwalkan akan dirilis berikutnya.
- Waktu pementasan — Snapshot dipentaskan setidaknya untuk jangka waktu ini dan kemudian memenuhi syarat untuk dirilis selama acara rilis snapshot.
- Frekuensi rilis - Lama waktu antara peristiwa rilis.

Di bagian daftar Snapshot, status snapshot adalah sebagai berikut:

- Dirilis - Snapshot bersifat publik dan tersedia untuk dilihat oleh pengguna dengan izin untuk produk ini.
- Rilis tertunda — Snapshot selesai atau dalam periode pementasan minimum wajib dan dijadwalkan untuk rilis berikutnya.
- Private - Snapshot dibuat sebelum aktivasi profil keamanan atau memiliki kesalahan validasi dan tidak terlihat oleh publik. Snapshot pribadi tetap hanya dalam visibilitas penjual.

Topik

- [Buat snapshot](#)
- [Lihat snapshot](#)
- [Ekspor snapshot](#)
- [Lihat snapshot terbaru yang dirilis](#)

- [Tunda rilis snapshot](#)
- [Ubah preferensi untuk daftar snapshot](#)

Buat snapshot

Untuk membuat snapshot untuk profil Anda, ikuti langkah-langkah ini. Anda dapat membuat maksimal 20 snapshot per hari.

1. Masuk ke AWS Management Console dan buka [AWS Marketplace konsol](#).
2. Pilih Wawasan Vendor.
3. Dari Vendor Insights, pilih produk.
4. Pada halaman profil produk, buka daftar Snapshot, dan pilih Buat snapshot baru.
5. Sebuah pesan memberi tahu Anda bahwa jadwal snapshot akan berubah. Pilih Buat.

Note

Jadwal snapshot berubah saat snapshot baru dibuat. Snapshot baru dijadwalkan untuk waktu yang sama dengan snapshot yang dibuat secara manual. Pesan ini termasuk jadwal baru.

Snapshot baru dibuat dalam waktu 30 menit dan ditambahkan ke daftar snapshot. Snapshot baru dibuat dengan status rilis Tertunda. Tidak ada yang dapat melihat snapshot baru hingga status berubah menjadi Dirilis.

Lihat snapshot

Untuk melihat snapshot untuk profil Anda, ikuti langkah-langkah ini.

1. Masuk ke AWS Management Console dan buka [AWS Marketplace konsol](#).
2. Pilih Wawasan Vendor.
3. Dari Vendor Insights, pilih produk.
4. Pada halaman profil produk, buka daftar Snapshot, dan pilih ID Snapshot dari snapshot yang ingin Anda lihat.
5. Setelah selesai, pilih Kembali untuk keluar dari tampilan snapshot.

Ekspor snapshot

Anda dapat mengekspor ke format JSON atau CSV. Untuk mengekspor snapshot, ikuti langkah-langkah ini.

1. Masuk ke AWS Management Console dan buka [AWS Marketplace konsol](#).
2. Pilih Wawasan Vendor.
3. Dari Vendor Insights, pilih produk.
4. Pada halaman profil produk, buka daftar Snapshot, dan pilih ID Snapshot dari snapshot yang ingin Anda ekspor.
5. Pilih Ekspor.
6. Dari daftar dropdown, pilih Unduh (JSON) atau Unduh (CSV).

Lihat snapshot terbaru yang dirilis

Snapshot terbaru yang dirilis adalah apa yang digunakan pengguna untuk melihat dan menilai kesehatan produk Anda. Penting untuk mengetahui apa yang ada dalam snapshot terbaru Anda yang dirilis untuk memastikan bahwa Anda menggambarkan produk Anda dengan informasi yang akurat. Untuk melihat snapshot terbaru untuk profil Anda, ikuti langkah-langkah ini.

1. Masuk ke AWS Management Console dan buka [AWS Marketplace konsol](#).
2. Pilih Wawasan Vendor.
3. Dari Vendor Insights, pilih produk.
4. Pada halaman profil produk, buka daftar Snapshot, dan pilih ID Snapshot dari snapshot yang ingin Anda lihat.
5. Pilih Lihat snapshot terbaru yang dirilis.
6. Setelah selesai, pilih Kembali untuk keluar dari tampilan snapshot.

Tunda rilis snapshot

Untuk menunda rilis snapshot ke profil Anda, Anda dapat menunda rilis snapshot untuk ID Snapshot tertentu.

1. Masuk ke AWS Management Console dan buka [AWS Marketplace konsol](#).

2. Pilih Wawasan Vendor.
3. Dari Vendor Insights, pilih produk.
4. Pada halaman profil produk, buka daftar Snapshot, dan pilih ID Snapshot dari snapshot yang ingin Anda tunda rilis.
5. Dari ringkasan Snapshot, pilih Tunda rilis snapshot.
6. Sebuah pesan memberi tahu Anda bahwa jadwal snapshot akan berubah. Pilih Tunda.

Pesan sukses muncul, menunjukkan bahwa Anda telah berhasil menunda rilis snapshot untuk produk ini.

Ubah preferensi untuk daftar snapshot

Setelah membuat snapshot, Anda dapat mengubah preferensi bagaimana snapshot dilihat dalam daftar Snapshot.

1. Masuk ke AWS Management Console dan buka [AWS Marketplace konsol](#).
2. Pilih Wawasan Vendor.
3. Dari Vendor Insights, pilih produk.
4. Pada halaman profil produk, buka daftar Snapshot, dan pilih ID Snapshot dari snapshot yang ingin Anda ubah.
5. Pilih ikon preferensi. Anda dapat menyesuaikan preferensi berikut untuk snapshot Anda:
 - Ukuran halaman - Pilih berapa banyak snapshot yang ingin Anda cantumkan di setiap halaman: 10 sumber daya, 20 sumber daya, atau 50 sumber daya per halaman.
 - Bungkus garis - Pilih opsi untuk membungkus garis untuk melihat seluruh catatan.
 - Format waktu - Pilih apakah Anda ingin Absolute, Relatif, atau ISO.
 - Kolom yang terlihat - Pilih opsi yang ingin Anda lihat untuk detail snapshot: ID Snapshot, Status, dan Tanggal dibuat.

Mengontrol akses di Wawasan AWS Marketplace Vendor

AWS Identity and Access Management (IAM) adalah Layanan AWS yang membantu Anda mengontrol akses ke AWS sumber daya. IAM adalah Layanan AWS yang dapat Anda gunakan tanpa biaya tambahan. Jika Anda seorang administrator, Anda mengontrol siapa yang dapat diautentikasi (masuk) dan diotorisasi (memiliki izin) untuk menggunakan AWS Marketplace sumber daya. AWS

Marketplace Vendor Insights menggunakan IAM untuk mengontrol akses ke data penjual, penilaian, pengesahan mandiri penjual, dan laporan audit standar industri.

Cara yang disarankan untuk mengontrol siapa yang dapat melakukan apa yang Portal Manajemen AWS Marketplace ada adalah dengan menggunakan IAM untuk membuat pengguna dan grup. Kemudian Anda menambahkan pengguna ke grup, dan mengelola grup. Anda dapat menetapkan kebijakan atau izin ke grup yang menyediakan izin hanya-baca. Jika Anda memiliki pengguna lain yang memerlukan akses hanya-baca, Anda dapat menambahkannya ke grup yang Anda buat daripada menambahkan izin untuk pengguna.

Kebijakan adalah dokumen yang menentukan izin yang berlaku untuk pengguna, grup, atau peran. Izin menentukan apa yang dapat dilakukan pengguna. AWS Kebijakan biasanya mengizinkan akses ke tindakan tertentu, dan secara opsional dapat memberikan bahwa tindakan tersebut diizinkan untuk sumber daya tertentu, seperti EC2 instans Amazon, bucket Amazon S3, dan sebagainya. Kebijakan juga dapat secara eksplisit menolak akses. Izin adalah pernyataan dalam kebijakan yang memungkinkan atau menolak akses ke sumber daya tertentu.

Important

Semua pengguna yang Anda buat mengautentikasi dengan menggunakan kredensialnya. Namun, mereka menggunakan hal yang sama Akun AWS. Setiap perubahan yang dilakukan pengguna dapat memengaruhi seluruh akun.

AWS Marketplace memiliki izin yang ditentukan untuk mengontrol tindakan yang dapat dilakukan seseorang dengan izin tersebut di Portal Manajemen AWS Marketplace. Ada juga kebijakan yang AWS Marketplace dibuat dan dikelola yang menggabungkan beberapa izin. `AWSMarketplaceSellerProductsFullAccess` Kebijakan ini memberi pengguna akses penuh ke produk di Portal Manajemen AWS Marketplace.

Untuk informasi selengkapnya tentang tindakan, sumber daya, dan kunci kondisi yang tersedia, lihat [Kunci tindakan, sumber daya, dan kondisi untuk Wawasan AWS Marketplace Vendor](#) di Referensi Otorisasi Layanan.

Izin untuk AWS Marketplace penjual Wawasan Vendor

Anda dapat menggunakan izin berikut dalam kebijakan IAM untuk Wawasan AWS Marketplace Vendor. Anda dapat menggabungkan izin ke dalam satu kebijakan IAM untuk memberikan izin yang Anda inginkan.

CreateDataSource

CreateDataSource memungkinkan pengguna untuk membuat sumber daya sumber data baru. Sumber data yang didukung adalah:

- SOC2Type2AuditReport
- ISO27001AuditReport
- AWSAuditManagerSecurityAutomatedAssessment
- FedRAMPCertification
- GDPRComplianceReport
- HIPAAComplianceReport
- PCIDSSAuditReport
- SecuritySelfAssessment

Grup aksi: Baca-tulis

Sumber daya yang dibutuhkan: Tidak ada

Menciptakan sumber daya: DataSource

DeleteDataSource

DeleteDataSource memungkinkan pengguna untuk menghapus sumber data yang mereka miliki. Sumber data harus dilepaskan dari profil apa pun yang akan dihapus. Untuk informasi selengkapnya, lihat [the section called “AssociateDataSource”](#).

Grup aksi: Baca-tulis

Sumber daya yang dibutuhkan: DataSource

GetDataSource

GetDataSource memungkinkan pengguna untuk mengambil rincian sumber data. Rincian sumber data mencakup informasi metadata seperti stempel waktu terkait, parameter pembuatan asli, dan informasi pemrosesan, jika ada.

Grup aksi: Hanya baca, baca-tulis

Sumber daya yang dibutuhkan: DataSource

UpdateDataSource

UpdateDataSource memungkinkan pengguna untuk memperbarui rincian sumber data. Detail mencakup informasi metadata, seperti nama dan informasi sumber (misalnya, peran, sumber Nama Sumber Daya Amazon (ARN), dan konten sumber).

Grup aksi: Hanya baca, baca-tulis

Sumber daya yang dibutuhkan: DataSource

ListDataSources

ListDataSources memungkinkan pengguna untuk membuat daftar sumber data yang mereka miliki.

Grup aksi: Read-only, read-write, list-only

Sumber daya yang dibutuhkan: Tidak ada

CreateSecurityProfile

CreateSecurityProfile memungkinkan pengguna untuk membuat profil keamanan baru. Profil keamanan adalah sumber daya untuk mengelola bagaimana dan kapan snapshot dihasilkan. Pengguna juga dapat mengontrol bagaimana pembeli dapat mengakses snapshot dengan mengontrol status dan ketentuan profil yang berlaku.

Grup aksi: Read-only, read-write, list-only

Sumber daya yang dibutuhkan: Tidak ada

Menciptakan sumber daya: SecurityProfile

ListSecurityProfiles

ListSecurityProfiles memungkinkan pengguna untuk membuat daftar profil keamanan yang mereka miliki.

Grup aksi: Read-only, read-write, list-only

Sumber daya yang dibutuhkan: Tidak ada

GetSecurityProfile

CreateSecurityProfilememungkinkan pengguna untuk mendapatkan detail profil keamanan.

Grup aksi: Hanya baca dan baca-tulis

Sumber daya yang dibutuhkan: SecurityProfile

AssociateDataSource

AssociateDataSourcემemungkinkan pengguna untuk mengaitkan yang sudah ada DataSource dengan profil AWS Marketplace Vendor Insights. Pengguna dapat mengontrol konten snapshot dengan mengaitkan atau memisahkan sumber data ke profil.

Grup aksi: Baca-tulis

Sumber daya yang dibutuhkan: SecurityProfile dan DataSource

DisassociateDataSource

DisassociateDataSourcემemungkinkan pengguna untuk memisahkan yang sudah ada DataSource dengan profil AWS Marketplace Vendor Insights. Pengguna dapat mengontrol konten snapshot dengan mengaitkan atau memisahkan sumber data ke profil.

Grup aksi: Baca-tulis

Sumber daya yang dibutuhkan: SecurityProfile dan DataSource

UpdateSecurityProfile

UpdateSecurityProfilememungkinkan pengguna untuk memodifikasi atribut profil keamanan seperti nama dan deskripsi.

Grup aksi: Baca-tulis

Sumber daya yang dibutuhkan: SecurityProfile

ActivateSecurityProfile

ActivateSecurityProfilememungkinkan pengguna untuk menetapkan Active status untuk profil keamanan. Setelah profil keamanan diaktifkan, snapshot baru

dapat dibuat dalam Staged keadaan yang memungkinkan untuk melepaskannya jika kondisi lain terpenuhi. Untuk informasi selengkapnya, lihat [the section called “UpdateSecurityProfileSnapshotReleaseConfiguration”](#).

Profil Active keamanan dengan setidaknya satu Released snapshot memenuhi syarat untuk penemuan Wawasan AWS Marketplace Vendor bagi pengguna akhir.

Grup aksi: Baca-tulis

Sumber daya yang dibutuhkan: SecurityProfile

DeactivateSecurityProfile

DeactivateSecurityProfilememungkinkan pengguna untuk menetapkan Inactive status untuk profil keamanan. Status terminal untuk profil keamanan ini setara dengan menghapus profil dari status bersama. Pengguna hanya dapat menonaktifkan profil keamanan jika tidak ada pelanggan aktif ke profil.

Grup aksi: Baca-tulis

Sumber daya yang dibutuhkan: SecurityProfile

UpdateSecurityProfileSnapshotCreationConfiguration

UpdateSecurityProfileSnapshotCreationConfigurationmemungkinkan pengguna untuk menentukan jadwal kustom untuk konfigurasi pembuatan snapshot. Konfigurasi pembuatan default pembuatan mingguan dapat diganti dengan tindakan ini.

Pengguna dapat menggunakan tindakan ini untuk mengubah jadwal termasuk membatalkan jadwal, menunda jadwal ke tanggal masa depan, atau memulai pembuatan snapshot baru untuk waktu yang lebih awal.

Grup aksi: Baca-tulis

Sumber daya yang dibutuhkan: SecurityProfile

UpdateSecurityProfileSnapshotReleaseConfiguration

UpdateSecurityProfileSnapshotReleaseConfigurationmemungkinkan pengguna untuk menentukan jadwal kustom untuk konfigurasi rilis snapshot. Konfigurasi pembuatan default rilis mingguan dengan periode pementasan dua hari untuk ditinjau dapat diganti dengan tindakan ini.

Pengguna dapat menggunakan tindakan ini untuk mengubah jadwal termasuk membatalkan jadwal atau menunda jadwal ke tanggal yang akan datang.

Grup aksi: Baca-tulis

Sumber daya yang dibutuhkan: SecurityProfile

ListSecurityProfileSnapshots

ListSecurityProfileSnapshots memungkinkan pengguna untuk membuat daftar snapshot untuk profil keamanan yang mereka miliki.

Grup aksi: Hanya baca, hanya daftar, dan baca-tulis

Sumber daya yang dibutuhkan: SecurityProfile

GetSecurityProfileSnapshot

GetSecurityProfileSnapshot memungkinkan pengguna untuk mendapatkan snapshot untuk profil keamanan yang mereka miliki.

Grup aksi: Hanya baca dan baca-tulis

Sumber daya yang dibutuhkan: SecurityProfile

TagResource

TagResource memungkinkan pengguna untuk menambahkan tag baru ke sumber daya. Sumber daya yang didukung adalah SecurityProfile dan DataSource.

Grup aksi: Menandai

Sumber daya opsional: SecurityProfile dan DataSource

UntagResource

UntagResource memungkinkan pengguna untuk menghapus tag dari sumber daya. Sumber daya yang didukung adalah SecurityProfile dan DataSource.

Grup aksi: Menandai

Sumber daya opsional: SecurityProfile dan DataSource

ListTagsForResource

ListTagsForResource memungkinkan pengguna untuk membuat daftar tag sumber daya untuk sumber daya. Sumber daya yang didukung adalah SecurityProfile dan DataSource.

Grup aksi: Hanya baca

Sumber daya opsional: SecurityProfile dan DataSource

Sumber daya tambahan

Sumber daya berikut dalam Panduan Pengguna IAM memberikan informasi lebih lanjut tentang memulai dan menggunakan IAM:

- [Praktik terbaik keamanan di IAM](#)
- [Mengelola kebijakan IAM](#)
- [Melampirkan kebijakan ke grup pengguna IAM](#)
- [Identitas IAM \(pengguna, grup pengguna, dan peran\)](#)
- [Buat pengguna IAM di akun Anda AWS](#)
- [Buat grup pengguna IAM](#)
- [Mengontrol akses ke AWS sumber daya menggunakan kebijakan](#)

AWS Marketplace keamanan

Keamanan cloud di AWS adalah prioritas tertinggi. Sebagai AWS pelanggan, Anda mendapat manfaat dari pusat data dan arsitektur jaringan yang dibangun untuk memenuhi persyaratan organisasi yang paling sensitif terhadap keamanan.

Keamanan adalah tanggung jawab bersama antara Anda AWS dan Anda. [Model tanggung jawab bersama](#) menggambarkan hal ini sebagai keamanan dari cloud dan keamanan di cloud:

- Keamanan cloud — AWS bertanggung jawab untuk melindungi infrastruktur yang menjalankan AWS layanan di AWS Cloud. AWS juga memberi Anda layanan yang dapat Anda gunakan dengan aman. Efektivitas keamanan kami diuji dan diverifikasi secara rutin oleh auditor pihak ketiga sebagai bagian dari [program kepatuhan AWS](#). Untuk mempelajari tentang program kepatuhan yang berlaku AWS Marketplace, lihat [AWS Layanan dalam Lingkup berdasarkan Program Kepatuhan](#).
- Keamanan di cloud — Tanggung jawab Anda ditentukan oleh AWS layanan yang Anda gunakan. Anda juga bertanggung jawab atas faktor-faktor lain termasuk sensitivitas data Anda, persyaratan organisasi Anda, dan hukum dan peraturan yang berlaku.

Dokumentasi ini membantu Anda memahami cara menerapkan model tanggung jawab bersama saat menggunakan AWS Marketplace. Topik berikut menunjukkan kepada Anda cara mengonfigurasi AWS Identity and Access Management untuk mengelola akses AWS Marketplace agar dapat memenuhi tujuan keamanan dan kepatuhan Anda. Anda juga dapat mempelajari cara menggunakan AWS layanan lain yang dapat membantu Anda memantau dan mengamankan AWS Marketplace sumber daya Anda.

Untuk mempelajari lebih lanjut tentang keamanan dan kebijakan lain mengenai produk yang Anda tawarkan AWS Marketplace, lihat topik berikut:

- [Persyaratan produk berbasis AMI untuk AWS Marketplace](#)
- [Persyaratan produk berbasis kontainer untuk AWS Marketplace](#)
- [Persyaratan dan praktik terbaik untuk membuat produk pembelajaran mesin](#)
- [Pedoman produk SaaS untuk AWS Marketplace](#)
- [Persyaratan untuk produk layanan profesional AWS Marketplace](#)

Note

Untuk mempelajari tentang keamanan di AWS Data Exchange untuk produk data, lihat [Keamanan](#) di Panduan Pengguna AWS Data Exchange.

Untuk mempelajari tentang keamanan bagi pembeli di AWS Marketplace, lihat [Keamanan AWS Marketplace di Panduan AWS Marketplace Pembeli](#).

Topik

- [Mengontrol akses ke Portal Manajemen AWS Marketplace](#)
- [Kebijakan dan izin untuk penjual AWS Marketplace](#)
- [AWS kebijakan terkelola untuk AWS Marketplace penjual](#)
- [AWS Marketplace Izin akun Layanan Analisis Perdagangan](#)
- [Izin Amazon SQS](#)
- [AWS Marketplace izin API pengukuran dan hak](#)
- [Menggunakan peran terkait layanan untuk Otorisasi Penjualan Kembali dengan AWS Marketplace](#)
- [Panggilan Logging AWS Marketplace Metering API dengan AWS CloudTrail](#)

Mengontrol akses ke Portal Manajemen AWS Marketplace

AWS Identity and Access Management (IAM) adalah Layanan AWS yang membantu Anda mengontrol akses ke AWS sumber daya. Jika Anda seorang administrator, Anda mengontrol siapa yang dapat diautentikasi (masuk) dan diberi wewenang (memiliki izin) untuk menggunakan AWS Marketplace sumber daya. IAM adalah Layanan AWS yang dapat Anda gunakan tanpa biaya tambahan.

Cara yang disarankan untuk mengontrol siapa yang dapat melakukan apa yang Portal Manajemen AWS Marketplace ada adalah dengan menggunakan IAM untuk membuat pengguna dan grup. Kemudian Anda menambahkan pengguna ke grup, dan mengelola grup. Misalnya, jika John diizinkan untuk melihat produk Anda, buat pengguna untuknya dan tambahkan penggunaanya ke grup yang Anda buat untuk akses hanya-baca. Anda dapat menetapkan kebijakan atau izin ke grup yang menyediakan izin hanya-baca. Jika Anda memiliki pengguna lain yang memerlukan akses hanya-baca, Anda dapat menambahkannya ke grup yang Anda buat daripada menambahkan izin ke pengguna. Jika peran John berubah dan dia tidak lagi membutuhkan akses hanya-baca, Anda dapat menghapus John dari grup.

Kebijakan adalah dokumen yang menentukan izin yang berlaku untuk pengguna, grup, atau peran. Pada gilirannya, izin menentukan apa yang dapat dilakukan pengguna. AWS Kebijakan biasanya mengizinkan akses ke tindakan tertentu, dan secara opsional dapat memberikan bahwa tindakan tersebut diizinkan untuk sumber daya tertentu, seperti EC2 instans Amazon, bucket Amazon S3, dan sebagainya. Kebijakan juga dapat secara eksplisit menolak akses. Izin adalah pernyataan dalam kebijakan yang memungkinkan atau menolak akses ke sumber daya tertentu. Anda dapat menyatakan izin apa pun seperti ini: “A memiliki izin untuk melakukan B ke C.” Misalnya, Jane (A) memiliki izin untuk membaca pesan (B) dari antrian Layanan Antrian Sederhana Amazon John (C). Setiap kali Jane mengirimkan permintaan ke Amazon SQS untuk menggunakan antrian John, layanan memeriksa untuk melihat apakah dia memiliki izin. Ini lebih lanjut memeriksa untuk melihat apakah permintaan memenuhi persyaratan yang ditentukan John dalam izin.

 Important

Semua pengguna yang Anda buat mengautentikasi dengan menggunakan kredensialnya. Namun, mereka menggunakan hal yang sama Akun AWS. Setiap perubahan yang dilakukan pengguna dapat memengaruhi seluruh akun.

AWS Marketplace memiliki izin yang ditentukan untuk mengontrol tindakan yang dapat dilakukan seseorang dengan izin tersebut. Portal Manajemen AWS Marketplace Ada juga kebijakan yang AWS Marketplace membuat dan mengelola yang menggabungkan beberapa izin.

Sumber daya berikut memberikan informasi lebih lanjut tentang memulai dan menggunakan IAM.

- [Buat pengguna administratif](#)
- [Praktik terbaik keamanan di IAM](#)
- [Mengelola kebijakan IAM](#)
- [Melampirkan kebijakan ke grup pengguna IAM](#)
- [Identitas IAM \(pengguna, grup, dan peran\)](#)
- [Mengontrol akses ke AWS sumber daya menggunakan kebijakan](#)

Topik berikut memberikan beberapa panduan tingkat tinggi untuk membuat pengguna dan grup, dan masuk sebagai pengguna.

Topik

- [Membuat pengguna](#)
- [Membuat atau menggunakan grup](#)
- [Masuk sebagai pengguna](#)

Membuat pengguna

Untuk memungkinkan orang di perusahaan Anda masuk Portal Manajemen AWS Marketplace, buat pengguna untuk setiap orang yang membutuhkan akses.

Untuk membuat pengguna

1. Masuk ke AWS Management Console dan buka konsol IAM di <https://console.aws.amazon.com/iam/>.
2. Di panel navigasi, di bawah Manajemen akses, pilih Pengguna, lalu pilih Buat pengguna.
3. Di kotak teks bernomor, masukkan nama untuk setiap pengguna yang ingin Anda buat.
4. Kosongkan kotak centang Hasilkan kunci akses untuk setiap pengguna dan kemudian pilih Buat.

Untuk menetapkan kata sandi untuk setiap pengguna yang baru saja Anda buat

1. Dalam daftar pengguna, pilih nama pengguna baru.
2. Pilih tab Security Credentials dan kemudian pilih Manage Password.
3. Pilih opsi untuk kata sandi yang dibuat secara otomatis atau kata sandi khusus. Secara opsional, untuk meminta pengguna memilih kata sandi baru pada login berikutnya, pilih kotak untuk Memerlukan pengguna untuk membuat kata sandi baru saat masuk berikutnya. Pilih Terapkan.
4. Pilih Unduh Kredensial untuk menyimpan kredensial masuk dan URL masuk khusus akun ke file nilai yang dipisahkan koma (CSV) di komputer Anda. Lalu, pilih Tutup.

Note

Untuk masuk dengan kredensial masuk yang baru saja Anda buat, pengguna harus menavigasi ke URL login khusus akun Anda. URL ini ada di file kredensial yang baru saja Anda unduh dan juga tersedia di konsol IAM. Untuk informasi selengkapnya, lihat [Cara pengguna IAM masuk ke Panduan Pengguna IAM Anda Akun AWS](#).

Tip

Buat kredensi masuk untuk diri Anda sendiri juga, meskipun Anda adalah pemiliknya. Akun AWS Ini adalah praktik terbaik yang direkomendasikan bagi semua orang untuk bekerja AWS Marketplace sebagai pengguna, bahkan pemilik akun. Untuk petunjuk tentang cara membuat pengguna untuk diri sendiri yang memiliki izin administratif, lihat [Membuat pengguna administratif di Panduan Pengguna IAM](#).

Membuat atau menggunakan grup

Setelah Anda membuat pengguna, membuat grup, membuat izin untuk mengakses halaman di Portal Manajemen AWS Marketplace, menambahkan izin tersebut ke grup, dan kemudian menambahkan pengguna ke grup.

Saat Anda menetapkan izin ke grup, Anda mengizinkan anggota grup tersebut untuk melakukan tindakan tertentu. Saat Anda menambahkan pengguna baru ke grup, pengguna tersebut secara otomatis mendapatkan izin yang ditetapkan ke grup. Sebuah grup dapat memiliki izin untuk lebih dari satu tindakan. Sebaiknya gunakan [kebijakan AWS Marketplace terkelola](#) daripada membuat kebijakan Anda sendiri.

Untuk menetapkan kebijakan terkelola AWS Marketplace untuk grup

1. Buka konsol IAM di <https://console.aws.amazon.com/iam/>.
2. Di panel navigasi, pilih Grup, lalu pilih grup tempat Anda ingin melampirkan kebijakan.
3. Pada halaman ringkasan untuk grup, di bawah tab Izin, pilih Lampirkan Kebijakan.
4. Pada halaman Lampirkan Kebijakan, di samping Filter: enterawsmarketplace.
5. Pilih kebijakan atau kebijakan yang ingin dilampirkan, lalu pilih Lampirkan Kebijakan.

Untuk membuat kebijakan dengan Portal Manajemen AWS Marketplace izin

1. Buka konsol IAM di <https://console.aws.amazon.com/iam/>.
2. Di panel navigasi, pilih Kebijakan, lalu pilih Buat Kebijakan.
3. Di samping Policy Generator, pilih Select.
4. Di halaman Edit Izin, lakukan hal berikut:
 - a. Untuk Efek, pilih Izinkan.

- b. Untuk AWS Layanan, pilih Portal Manajemen AWS Marketplace.
 - c. Untuk Tindakan, pilih izin atau izin yang akan diizinkan.
 - d. Pilih Tambah Pernyataan.
 - e. Pilih Langkah Selanjutnya.
5. Pada halaman Kebijakan Tinjauan, lakukan hal berikut:
- a. Untuk Nama Kebijakan, masukkan nama untuk kebijakan ini. Catat nama kebijakan karena Anda membutuhkannya untuk langkah selanjutnya.
 - b. (Opsional) Untuk Deskripsi, masukkan deskripsi untuk kebijakan ini.
 - c. Pilih Buat Kebijakan.

Untuk membuat grup IAM dengan izin yang sesuai dan menambahkan pengguna ke grup

1. Buka konsol IAM di <https://console.aws.amazon.com/iam/>.
2. Di panel navigasi, pilih Grup lalu pilih Buat Grup Baru.
3. Untuk Nama Grup:, ketik nama untuk grup. Kemudian pilih Next Step.
4. Pada halaman Lampirkan Kebijakan, lakukan hal berikut:
 - a. Untuk Filter:, pilih Kebijakan yang Dikelola Pelanggan.
 - b. Pilih kotak centang di samping nama kebijakan yang ingin dilampirkan ke grup ini. Ini biasanya kebijakan yang baru saja Anda buat.
 - c. Pilih Langkah Selanjutnya.
5. Pilih Buat group.
6. Temukan grup baru Anda di daftar Grup, lalu pilih kotak centang di sebelahnya. Pilih Tindakan Grup dan kemudian Tambahkan Pengguna ke Grup.
7. Pilih kotak centang di samping setiap pengguna untuk ditambahkan ke grup dan kemudian pilih Tambah Pengguna.

Masuk sebagai pengguna

Setelah Anda membuat pengguna di IAM, pengguna dapat masuk dengan kredensialnya sendiri. Untuk melakukannya, mereka perlu menggunakan URL unik yang terkait dengan Anda Akun AWS. Anda bisa mendapatkan dan mendistribusikan URL masuk ke pengguna Anda.

Untuk mendapatkan URL masuk unik akun

1. Buka konsol IAM di <https://console.aws.amazon.com/iam/>.
2. Di panel navigasi, pilih Dasbor.
3. Di dekat bagian atas panel konten, temukan tautan masuk pengguna IAM: dan perhatikan tautan masuk, yang memiliki format seperti ini:

```
https://AWS_account_ID.signin.aws.amazon.com/console/
```

Note

Jika Anda ingin URL untuk halaman login berisi nama perusahaan Anda (atau pengenal ramah lainnya) alih-alih Akun AWS ID Anda, Anda dapat membuat alias untuk akun Anda dengan memilih Kustomisasi. Untuk informasi selengkapnya, lihat [ID AWS Akun Anda dan Aliasnya](#) di Panduan Pengguna IAM.

4. Bagikan URL ini kepada orang-orang di perusahaan Anda yang dapat bekerja sama AWS Marketplace, bersama dengan kredensi masuk yang Anda buat untuk masing-masing. Instruksikan mereka untuk menggunakan URL masuk unik akun Anda untuk masuk sebelum mereka mengakses. AWS Marketplace

Kebijakan dan izin untuk penjual AWS Marketplace

AWS Marketplace menyediakan seperangkat kebijakan terkelola untuk digunakan dengan Portal Manajemen AWS Marketplace. Selain itu, Anda dapat menggunakan izin individual untuk membuat kebijakan AWS Identity and Access Management (IAM) Anda sendiri.

Anda juga dapat memberikan akses halus ke tab Pengaturan, Hubungi Kami, Unggah File, dan Wawasan. Portal Manajemen AWS Marketplace Akses berbutir halus memungkinkan Anda melakukan hal berikut:

- Berikan izin kepada orang lain untuk mengelola dan menggunakan sumber daya di Akun AWS tanpa membagikan kata sandi atau kunci akses Anda.
- Berikan izin terperinci kepada banyak orang untuk berbagai sumber daya. Misalnya, Anda mungkin mengizinkan akses beberapa pengguna untuk melihat tab Pengaturan di Portal Manajemen AWS Marketplace. Untuk pengguna lain, Anda mungkin mengizinkan akses untuk mengedit di tab Pengaturan dan Hubungi Kami.

Note

Untuk informasi selengkapnya tentang kebijakan dan izin di AWS Data Exchange untuk produk data, lihat [Identity and Access Management di AWS Data Exchange](#) di Panduan Pengguna AWS Data Exchange.

Untuk informasi selengkapnya tentang kebijakan dan izin bagi AWS Marketplace pembeli, lihat [Mengontrol akses ke AWS Marketplace langganan di Panduan AWS Marketplace Pembeli](#).

Kebijakan untuk AWS Marketplace penjual

Anda dapat menggunakan kebijakan terkelola berikut untuk memberi pengguna akses terkontrol ke Portal Manajemen AWS Marketplace:

AWSMarketplaceSellerFullAccess

Memungkinkan akses penuh ke semua halaman di Portal Manajemen AWS Marketplace dan AWS layanan lainnya, seperti manajemen Amazon Machine Image (AMI).

AWSMarketplaceSellerProductsFullAccess

Memungkinkan akses penuh ke halaman [Produk](#) di Portal Manajemen AWS Marketplace.

AWSMarketplaceSellerProductsReadOnly

Memungkinkan akses hanya-baca ke halaman [Produk](#) di Portal Manajemen AWS Marketplace

Important

AWS Marketplace pembeli dapat menggunakan kebijakan terkelola untuk mengelola langganan yang mereka beli. Nama-nama kebijakan terkelola yang Anda gunakan Portal Manajemen AWS Marketplace dimulai dengan `AWSMarketplaceSeller`. Saat Anda mencari kebijakan di IAM, pastikan untuk mencari nama kebijakan yang dimulai dengan `AWSMarketplaceSeller`. Untuk informasi selengkapnya tentang kebijakan tersebut, lihat Referensi Kebijakan AWS Terkelola.

AWS Marketplace juga menyediakan kebijakan terkelola khusus untuk skenario tertentu. Untuk daftar lengkap kebijakan AWS terkelola bagi AWS Marketplace penjual dan deskripsi izin apa yang mereka berikan, lihat. [AWS kebijakan terkelola untuk AWS Marketplace penjual](#)

Izin untuk penjual AWS Marketplace

Anda dapat menggunakan izin berikut dalam kebijakan IAM untuk: Portal Manajemen AWS Marketplace

aws-marketplace-management:PutSellerVerificationDetails

Memungkinkan akses untuk memulai proses Know Your Customer (KYC).

aws-marketplace-management:GetSellerVerificationDetails

Memungkinkan akses untuk melihat status KYC di file. Portal Manajemen AWS Marketplace

aws-marketplace-management:PutBankAccountVerificationDetails

Memungkinkan akses untuk memulai proses [verifikasi rekening bank](#).

aws-marketplace-management:GetBankAccountVerificationDetails

Memungkinkan akses untuk melihat status verifikasi rekening bank di Portal Manajemen AWS Marketplace.

aws-marketplace-management:PutSecondaryUserVerificationDetails

Memungkinkan akses untuk menambahkan pengguna sekunder di file Portal Manajemen AWS Marketplace.

aws-marketplace-management:GetSecondaryUserVerificationDetails

Memungkinkan akses untuk melihat status pengguna sekunder di file Portal Manajemen AWS Marketplace.

aws-marketplace-management:GetAdditionalSellerNotificationRecipients

Memungkinkan akses untuk melihat kontak email untuk AWS Marketplace pemberitahuan.

aws-marketplace-management:PutAdditionalSellerNotificationRecipients

Memungkinkan akses untuk memperbarui kontak email untuk AWS Marketplace pemberitahuan.

tax:PutTaxInterview

Memungkinkan akses untuk mengambil [wawancara pajak](#) di Portal Manajemen AWS Marketplace.

tax:GetTaxInterview

Memungkinkan akses untuk melihat status wawancara pajak di Portal Manajemen AWS Marketplace.

tax:GetTaxInfoReportingDocument

Memungkinkan AWS Marketplace penjual untuk melihat dan mengunduh dokumen pajak (misalnya, formulir 1099-K) dari dasbor Pajak

payments:CreatePaymentInstrument

Memungkinkan akses untuk menambahkan rekening bank ke Portal Manajemen AWS Marketplace.

payments:GetPaymentInstrument

Memungkinkan akses untuk melihat rekening bank yang ada di Portal Manajemen AWS Marketplace.

support:CreateCase

Memungkinkan akses untuk membuat AWS Marketplace kasus di dalam file Portal Manajemen AWS Marketplace.

aws-marketplace-management:viewSupport

Memungkinkan akses ke halaman [Customer Support Eligibility](#) di halaman. Portal Manajemen AWS Marketplace

aws-marketplace-management:viewReports

Memungkinkan akses ke halaman [Laporan](#) di Portal Manajemen AWS Marketplace.

aws-marketplace>ListEntities

Memungkinkan akses ke daftar objek di Portal Manajemen AWS Marketplace. Diperlukan untuk mengakses halaman [Unggah File](#), [Penawaran](#), dan [Mitra](#) di halaman Portal Manajemen AWS Marketplace.

Note

Untuk mengizinkan akses melihat tab Pengaturan, Anda dapat menggunakan izin ini, ListEntity izin, dan Nama Sumber Daya Amazon (ARN) berikut.:

```
arn:{partition}:{aws-marketplace}:{region}:{account-id}:AWSMarketplace/Seller/{entity-id}
```

aws-marketplace:DescribeEntity

Memungkinkan akses untuk melihat detail objek di Portal Manajemen AWS Marketplace. Diperlukan untuk mengakses halaman [Unggah File](#), [Penawaran](#), [Mitra](#), dan [Perjanjian](#) di halaman Portal Manajemen AWS Marketplace.

Note

Untuk mengizinkan akses untuk melihat tab Pengaturan, Anda dapat menggunakan izin ini, DescribeEntity izin, dan ARN berikut: `arn:{partition}:{aws-marketplace}:{region}:{account-id}:AWSMarketplace/Seller/*`

aws-marketplace:StartChangeSet

Memungkinkan akses untuk membuat perubahan produk di Portal Manajemen AWS Marketplace. Diperlukan untuk membuat perubahan pada halaman [Unggah File](#), [Penawaran](#), [Mitra](#), dan [Perjanjian](#) di halaman Portal Manajemen AWS Marketplace.

Note

Untuk mengizinkan akses mendaftar sebagai penjual AWS Marketplace, Anda dapat menggunakan izin ini, kunci `catalog:ChangeType: "CreateSeller"` kondisi, dan ARN berikut: `arn:{partition}:{aws-marketplace}:{region}:{account-id}:AWSMarketplace/Seller/{entity-id}`

Untuk mengizinkan akses memperbarui profil penjual AWS Marketplace, Anda dapat menggunakan izin ini, kunci `catalog:ChangeType: "UpdateInformation"` kondisi, dan ARN berikut: `arn:{partition}:{aws-marketplace}:{region}:{account-id}:AWSMarketplace/Seller/{entity-id}`

Untuk mengizinkan akses memperbarui preferensi pencairan untuk Amazon Web Services, Anda dapat menggunakan izin ini, kunci `catalog:ChangeType: "UpdateDisbursementPreferences"` kondisi, dan ARN berikut: `arn:{partition}:{aws-marketplace}:{region}:{account-id}:AWSMarketplace/Seller/{entity-id}`

aws-marketplace:SearchAgreements

Memungkinkan melihat daftar perjanjian tingkat tinggi di halaman [Perjanjian](#), dan peluang antara ISVs dan mitra saluran di halaman [Mitra](#).

aws-marketplace:DescribeAgreement

Memungkinkan melihat detail perjanjian tingkat tinggi di halaman Perjanjian, dan peluang antara ISVs dan mitra saluran di halaman Mitra.

aws-marketplace:GetAgreementTerms

Memungkinkan melihat semua detail jangka waktu perjanjian di halaman Perjanjian, dan peluang antara ISVs dan mitra saluran di halaman Mitra.

aws-marketplace:GetSellerDashboard

Memungkinkan akses ke dasbor di halaman Wawasan di Portal Manajemen AWS Marketplace

aws-marketplace>ListAssessments

Memungkinkan akses untuk melihat daftar penilaian tindakan penjual yang tertunda.

aws-marketplace:DescribeAssessment

Memungkinkan akses untuk melihat detail penilaian tindakan penjual yang tertunda.

Note

Untuk memungkinkan pengguna mengakses halaman [Kelola Produk](#), Anda harus menggunakan izin `AWSMarketplaceSellerProductsFullAccess` atau `AWSMarketplaceSellerProductsReadOnly` terkelola.

Anda dapat menggabungkan izin sebelumnya ke dalam satu kebijakan IAM untuk memberikan izin yang Anda inginkan. Lihat contoh berikut.

Contoh 1: Izin untuk melihat status KYC

Untuk memberikan izin untuk melihat status KYC di Portal Manajemen AWS Marketplace, gunakan kebijakan yang mirip dengan contoh berikut.

Untuk memberikan izin untuk melihat status KYC di Portal Manajemen AWS Marketplace, gunakan kebijakan yang mirip dengan contoh berikut.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "aws-marketplace-management:GetSellerVerificationDetails"
      ],
      "Resource": ["*"]
    }
  ]
}
```

Contoh 2: Izin untuk membuat upgrade dan perpanjangan untuk penawaran pribadi

Untuk memberikan izin untuk melihat dan menggunakan halaman Perjanjian untuk membuat peningkatan dan perpanjangan untuk penawaran pribadi, gunakan kebijakan yang mirip dengan contoh berikut.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "aws-marketplace:SearchAgreements",
        "aws-marketplace:DescribeAgreement",
        "aws-marketplace:GetAgreementTerms",
        "aws-marketplace:DescribeEntity",
        "aws-marketplace:StartChangeSet"
      ],
      "Effect": "Allow",
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "aws-marketplace:PartyType": "Proposer"
        }
      }
    }
  ]
}
```

```

        "ForAllValues:StringEquals": {
            "aws-marketplace:AgreementType": [
                "PurchaseAgreement"
            ]
        }
    ]
}

```

Contoh 3: Izin untuk mengakses halaman Penawaran dan membuat penawaran pribadi baru

Untuk memberikan izin untuk melihat dan menggunakan halaman Penawaran untuk melihat penawaran pribadi yang ada dan membuat penawaran pribadi, gunakan kebijakan yang mirip dengan contoh berikut.

JSON

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "aws-marketplace:ListEntities",
        "aws-marketplace:DescribeEntity",
        "aws-marketplace:StartChangeSet"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}

```

Contoh 4: Izin untuk mengakses halaman Pengaturan

Untuk memberikan izin untuk melihat dan menggunakan halaman Pengaturan, gunakan kebijakan yang mirip dengan contoh berikut.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "aws-marketplace:ListEntities",
        "aws-marketplace:DescribeEntity",
        "aws-marketplace:StartChangeSet"
      ],
      "Effect": "Allow",
      "Resource": "arn:{partition}:{aws-marketplace}:{region}:{account-
id}:AWSMarketplace/Seller/*",
    }
  ]
}
```

Contoh 5: Izin untuk mengakses halaman Unggah File

Untuk memberikan izin untuk melihat dan menggunakan halaman Unggah File, gunakan kebijakan yang mirip dengan contoh berikut.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "aws-marketplace:ListEntities",
        "aws-marketplace:DescribeEntity",
        "aws-marketplace:StartChangeSet"
      ],
      "Effect": "Allow",
      "Resource": "*",
    }
  ]
}
```

Menggunakan grup IAM

Atau, Anda dapat membuat grup IAM terpisah untuk memberikan akses ke setiap halaman individual di Portal Manajemen AWS Marketplace Pengguna dapat menjadi anggota lebih dari satu grup. Jadi, jika pengguna membutuhkan akses ke lebih dari satu halaman, Anda dapat menambahkan

pengguna ke semua grup yang sesuai. Misalnya, buat satu grup IAM dan berikan izin grup tersebut untuk mengakses halaman Wawasan, membuat grup lain, dan memberikan izin grup tersebut untuk mengakses halaman Unggah File, dan seterusnya. Jika pengguna memerlukan izin untuk mengakses halaman Wawasan dan halaman Unggah File, tambahkan pengguna ke kedua grup.

Untuk informasi selengkapnya tentang pengguna dan grup, lihat [Identitas IAM \(pengguna, grup, dan peran\)](#) di Panduan Pengguna IAM.

AWS kebijakan terkelola untuk AWS Marketplace penjual

Kebijakan AWS terkelola adalah kebijakan mandiri yang dibuat dan dikelola oleh AWS. AWS Kebijakan terkelola dirancang untuk memberikan izin bagi banyak kasus penggunaan umum sehingga Anda dapat mulai menetapkan izin kepada pengguna, grup, dan peran.

Perlu diingat bahwa kebijakan AWS terkelola mungkin tidak memberikan izin hak istimewa paling sedikit untuk kasus penggunaan spesifik Anda karena tersedia untuk digunakan semua pelanggan. AWS Kami menyarankan Anda untuk mengurangi izin lebih lanjut dengan menentukan [kebijakan yang dikelola pelanggan](#) yang khusus untuk kasus penggunaan Anda.

Anda tidak dapat mengubah izin yang ditentukan dalam kebijakan AWS terkelola. Jika AWS memperbarui izin yang ditentukan dalam kebijakan AWS terkelola, pembaruan akan memengaruhi semua identitas utama (pengguna, grup, dan peran) yang dilampirkan kebijakan tersebut. AWS kemungkinan besar akan memperbarui kebijakan AWS terkelola saat baru Layanan AWS diluncurkan atau operasi API baru tersedia untuk layanan yang ada.

Untuk informasi selengkapnya, lihat [Kebijakan terkelola AWS](#) dalam Panduan Pengguna IAM.

Bagian ini mencantumkan setiap kebijakan yang digunakan untuk mengelola akses penjual AWS Marketplace. Untuk informasi tentang kebijakan pembeli, lihat [kebijakan AWS terkelola untuk AWS MarketplaceAWS Marketplace pembeli](#) di Panduan Pembeli.

Topik

- [AWS kebijakan terkelola: AWSMarketplaceAmiIngestion](#)
- [AWS kebijakan terkelola: AWSMarketplaceFullAccess](#)
- [AWS kebijakan terkelola: AWSMarketplaceGetEntitlements](#)
- [AWS kebijakan terkelola: AWSMarketplaceMeteringFullAccess](#)
- [Kebijakan terkelola AWS : AWSMarketplaceMeteringRegisterUsage](#)

- [AWS kebijakan terkelola: AWSMarketplaceSellerFullAccess](#)
- [Kebijakan terkelola AWS : AWSMarketplaceSellerProductsFullAccess](#)
- [Kebijakan terkelola AWS : AWSMarketplaceSellerProductsReadOnly](#)
- [Kebijakan terkelola AWS : AWSMarketplaceSellerOfferManagement](#)
- [AWS kebijakan terkelola: AWSVendorInsightsVendorFullAccess](#)
- [AWS kebijakan terkelola: AWSVendorInsightsVendorReadOnly](#)
- [AWS Marketplace pembaruan kebijakan AWS terkelola](#)

AWS kebijakan terkelola: AWSMarketplaceAmiIngestion

Anda dapat membuat peran layanan dengan kebijakan ini yang kemudian dapat digunakan oleh AWS Marketplace untuk melakukan tindakan atas nama Anda. Untuk informasi selengkapnya tentang penggunaan AWSMarketplaceAmiIngestion, lihat [Memberikan AWS Marketplace akses ke AMI Anda](#).

Kebijakan ini memberikan izin kontributor yang memungkinkan AWS Marketplace untuk menyalin Gambar Mesin Amazon (AMIs) Anda untuk mencantulkannya. AWS Marketplace

Untuk melihat izin kebijakan ini, lihat [AWSMarketplaceAmiIngestion](#) di Referensi Kebijakan AWS Terkelola.

AWS kebijakan terkelola: AWSMarketplaceFullAccess

Anda dapat melampirkan kebijakan AWSMarketplaceFullAccess ke identitas IAM Anda.

Kebijakan ini memberikan izin administratif yang memungkinkan akses penuh ke AWS Marketplace dan layanan terkait, baik sebagai penjual maupun pembeli. Izin ini mencakup kemampuan berikut:

- Berlangganan dan berhenti berlangganan AWS Marketplace perangkat lunak.
- Kelola instance AWS Marketplace perangkat lunak dari AWS Marketplace.
- Buat dan kelola pasar pribadi di akun Anda.
- Menyediakan akses ke Amazon EC2, AWS CloudFormation, dan Amazon EC2 Systems Manager.

Untuk melihat izin kebijakan ini, lihat [AWSMarketplaceFullAccess](#) di Referensi Kebijakan AWS Terkelola.

AWS kebijakan terkelola: `AWSMarketplaceGetEntitlements`

Anda dapat melampirkan kebijakan `AWSMarketplaceGetEntitlements` ke identitas IAM Anda.

Kebijakan ini memberikan izin hanya-baca yang memungkinkan penjual produk perangkat lunak sebagai layanan (SaaS) untuk memeriksa apakah pelanggan telah berlangganan produk SaaS mereka. AWS Marketplace

Untuk melihat izin kebijakan ini, lihat [AWSMarketplaceGetEntitlements](#) di Referensi Kebijakan AWS Terkelola.

AWS kebijakan terkelola: `AWSMarketplaceMeteringFullAccess`

Anda dapat melampirkan kebijakan `AWSMarketplaceMeteringFullAccess` ke identitas IAM Anda.

Kebijakan ini memberikan izin kontributor yang memungkinkan pelaporan penggunaan terukur yang sesuai dengan AMI dan produk kontainer dengan harga konsumsi yang fleksibel. AWS Marketplace

Untuk melihat izin kebijakan ini, lihat [AWSMarketplaceMeteringFullAccess](#) di Referensi Kebijakan AWS Terkelola.

Kebijakan terkelola AWS : `AWSMarketplaceMeteringRegisterUsage`

Anda dapat melampirkan kebijakan `AWSMarketplaceMeteringRegisterUsage` ke identitas IAM Anda.

Kebijakan ini memberikan izin kontributor yang memungkinkan pelaporan penggunaan terukur yang sesuai dengan produk kontainer dengan harga per jam. AWS Marketplace

Untuk melihat izin kebijakan ini, lihat [AWSMarketplaceMeteringRegisterUsage](#) di Referensi Kebijakan AWS Terkelola.

AWS kebijakan terkelola: `AWSMarketplaceSellerFullAccess`

Anda dapat melampirkan kebijakan `AWSMarketplaceSellerFullAccess` ke identitas IAM Anda.

Kebijakan ini memberikan izin administratif yang memungkinkan akses penuh ke semua operasi penjual AWS Marketplace, termasuk Portal Manajemen AWS Marketplace, dan mengelola AMI Amazon yang digunakan dalam produk berbasis EC2 AMI.

JSON

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "MarketplaceManagement",
      "Effect": "Allow",
      "Action": [
        "aws-marketplace-management:uploadFiles",
        "aws-marketplace-management:viewReports",
        "aws-marketplace-management:viewSupport",
        "aws-marketplace:ListChangeSets",
        "aws-marketplace:DescribeChangeSet",
        "aws-marketplace:StartChangeSet",
        "aws-marketplace:CancelChangeSet",
        "aws-marketplace:ListEntities",
        "aws-marketplace:DescribeEntity",
        "aws-marketplace:GetSellerDashboard",
        "aws-marketplace:ListAssessments",
        "aws-marketplace:DescribeAssessment",
        "ec2:DescribeImages",
        "ec2:DescribeSnapshots",
        "ec2:ModifyImageAttribute",
        "ec2:ModifySnapshotAttribute"
      ],
      "Resource": "*"
    },
    {
      "Sid": "AgreementAccess",
      "Effect": "Allow",
      "Action": [
        "aws-marketplace:SearchAgreements",
        "aws-marketplace:DescribeAgreement",
        "aws-marketplace:GetAgreementTerms"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "aws-marketplace:PartyType": "Proposer"
        },
        "ForAllValues:StringEquals": {
          "aws-marketplace:AgreementType": [

```

```

        "PurchaseAgreement"
    ]
    }
}
},
{
    "Sid": "IAMGetRole",
    "Effect": "Allow",
    "Action": [
        "iam:GetRole"
    ],
    "Resource": "arn:aws:iam::*:role/*"
},
{
    "Sid": "AssetScanning",
    "Effect": "Allow",
    "Action": [
        "iam:PassRole"
    ],
    "Resource": "arn:aws:iam::*:role/*",
    "Condition": {
        "StringEquals": {
            "iam:PassedToService": "assets.marketplace.amazonaws.com"
        }
    }
},
{
    "Sid": "VendorInsights",
    "Effect": "Allow",
    "Action": [
        "vendor-insights:GetDataSource",
        "vendor-insights:ListDataSources",
        "vendor-insights:ListSecurityProfiles",
        "vendor-insights:GetSecurityProfile",
        "vendor-insights:GetSecurityProfileSnapshot",
        "vendor-insights:ListSecurityProfileSnapshots"
    ],
    "Resource": "*"
},
{
    "Sid": "TagManagement",
    "Effect": "Allow",
    "Action": [
        "aws-marketplace:TagResource",

```

```

        "aws-marketplace:UntagResource",
        "aws-marketplace:ListTagsForResource"
    ],
    "Resource": "arn:aws:aws-marketplace:*:*:AWSMarketplace/*"
},
{
    "Sid": "SellerSettings",
    "Effect": "Allow",
    "Action": [
        "aws-marketplace-management:GetSellerVerificationDetails",
        "aws-marketplace-management:PutSellerVerificationDetails",
        "aws-marketplace-management:GetBankAccountVerificationDetails",
        "aws-marketplace-management:PutBankAccountVerificationDetails",
        "aws-marketplace-management:GetSecondaryUserVerificationDetails",
        "aws-marketplace-management:PutSecondaryUserVerificationDetails",
        "aws-marketplace-
management:GetAdditionalSellerNotificationRecipients",
        "aws-marketplace-
management:PutAdditionalSellerNotificationRecipients",
        "payments:GetPaymentInstrument",
        "payments:CreatePaymentInstrument",
        "tax:GetTaxInterview",
        "tax:PutTaxInterview",
        "tax:GetTaxInfoReportingDocument",
        "tax:ListSupplementalTaxRegistrations",
        "tax:PutSupplementalTaxRegistration",
        "tax>DeleteSupplementalTaxRegistration",
        "tax:GetTaxRegistration"
    ],
    "Resource": "*"
},
{
    "Sid": "Support",
    "Effect": "Allow",
    "Action": [
        "support:CreateCase"
    ],
    "Resource": "*"
},
{
    "Sid": "ResourcePolicyManagement",
    "Effect": "Allow",
    "Action": [
        "aws-marketplace:GetResourcePolicy",

```

```

        "aws-marketplace:PutResourcePolicy",
        "aws-marketplace>DeleteResourcePolicy"
    ],
    "Resource": "arn:aws:aws-marketplace:*:*:AWSMarketplace/*"
  },
  {
    "Sid": "CreateServiceLinkedRole",
    "Effect": "Allow",
    "Action": "iam:CreateServiceLinkedRole",
    "Resource": "*",
    "Condition": {
      "StringEquals": {
        "iam:AWSServiceName": "resale-
authorization.marketplace.amazonaws.com"
      }
    }
  }
]
}

```

Untuk melihat izin kebijakan ini, lihat [AWSMarketplaceSellerFullAccess](#) di Referensi Kebijakan AWS Terkelola.

Kebijakan terkelola AWS : AWSMarketplaceSellerProductsFullAccess

Anda dapat melampirkan kebijakan AWSMarketplaceSellerProductsFullAccess ke identitas IAM Anda.

Kebijakan ini memberikan izin kontributor yang memungkinkan akses penuh untuk mengelola produk dan ke Portal Manajemen AWS Marketplace, serta mengelola Amazon AMI yang digunakan dalam produk berbasis EC2 AMI.

Detail izin

JSON

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",

```

```

    "Action": [
      "aws-marketplace:ListChangeSets",
      "aws-marketplace:DescribeChangeSet",
      "aws-marketplace:StartChangeSet",
      "aws-marketplace:CancelChangeSet",
      "aws-marketplace:ListEntities",
      "aws-marketplace:DescribeEntity",
      "aws-marketplace:ListAssessments",
      "aws-marketplace:DescribeAssessment",
      "ec2:DescribeImages",
      "ec2:DescribeSnapshots",
      "ec2:ModifyImageAttribute",
      "ec2:ModifySnapshotAttribute"
    ],
    "Resource": "*"
  },
  {
    "Effect": "Allow",
    "Action": [
      "iam:GetRole"
    ],
    "Resource": "arn:aws:iam::*:role/*"
  },
  {
    "Effect": "Allow",
    "Action": [
      "iam:PassRole"
    ],
    "Resource": "arn:aws:iam::*:role/*",
    "Condition": {
      "StringEquals": {
        "iam:PassedToService": "assets.marketplace.amazonaws.com"
      }
    }
  },
  {
    "Effect": "Allow",
    "Action": [
      "vendor-insights:GetDataSource",
      "vendor-insights:ListDataSources",
      "vendor-insights:ListSecurityProfiles",
      "vendor-insights:GetSecurityProfile",
      "vendor-insights:GetSecurityProfileSnapshot",
      "vendor-insights:ListSecurityProfileSnapshots"
    ]
  }
}

```

```

    ],
    "Resource": "*"
  },
  {
    "Effect": "Allow",
    "Action": [
      "aws-marketplace:TagResource",
      "aws-marketplace:UntagResource",
      "aws-marketplace:ListTagsForResource"
    ],
    "Resource": "arn:aws:aws-marketplace:*:*:AWSMarketplace/*"
  },
  {
    "Effect": "Allow",
    "Action": [
      "aws-marketplace:GetResourcePolicy",
      "aws-marketplace:PutResourcePolicy",
      "aws-marketplace>DeleteResourcePolicy"
    ],
    "Resource": "arn:aws:aws-marketplace:*:*:AWSMarketplace/*"
  }
]
}

```

Untuk melihat izin kebijakan ini, lihat [AWSMarketplaceSellerProductsFullAccess](#) di Referensi Kebijakan AWS Terkelola.

Kebijakan terkelola AWS : AWSMarketplaceSellerProductsReadOnly

Anda dapat melampirkan kebijakan `AWSMarketplaceSellerProductsReadOnly` ke identitas IAM Anda.

Kebijakan ini memberikan izin hanya-baca yang memungkinkan akses untuk melihat produk di Portal Manajemen AWS Marketplace, dan melihat Amazon AMI yang digunakan dalam produk berbasis EC2 AMI.

Detail izin

JSON

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "aws-marketplace:ListChangeSets",
        "aws-marketplace:DescribeChangeSet",
        "aws-marketplace:ListEntities",
        "aws-marketplace:DescribeEntity",
        "aws-marketplace:ListAssessments",
        "aws-marketplace:DescribeAssessment",
        "ec2:DescribeImages",
        "ec2:DescribeSnapshots"
      ],
      "Resource": "*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "aws-marketplace:ListTagsForResource"
      ],
      "Resource": "arn:aws:aws-marketplace:*:*:AWSMarketplace/*"
    },
    {
      "Effect": "Allow",
      "Action": [
        "aws-marketplace:GetResourcePolicy"
      ],
      "Resource": "arn:aws:aws-marketplace:*:*:AWSMarketplace/*"
    }
  ]
}

```

Untuk melihat izin kebijakan ini, lihat [AWSMarketplaceSellerProductsReadOnly](#) di Referensi Kebijakan AWS Terkelola.

Kebijakan terkelola AWS : AWSMarketplaceSellerOfferManagement

Anda dapat melampirkan kebijakan AWSMarketplaceSellerOfferManagement ke identitas IAM Anda.

Kebijakan ini memberi penjual akses ke aktivitas manajemen Penawaran dan Perjanjian.

Untuk melihat izin kebijakan ini, lihat [AWSMarketplaceSellerOfferManagement](#) di Referensi Kebijakan AWS Terkelola.

AWS kebijakan terkelola: AWSVendorInsightsVendorFullAccess

Anda dapat melampirkan kebijakan AWSVendorInsightsVendorFullAccess ke identitas IAM Anda.

Kebijakan ini memberikan akses penuh untuk membuat dan mengelola semua sumber daya di Wawasan AWS Marketplace Vendor. Dalam Wawasan AWS Marketplace Vendor, penilai sama dengan pembeli, dan vendor sama dengan penjual untuk keperluan panduan ini.

Untuk melihat izin kebijakan ini, lihat [AWSVendorInsightsVendorFullAccess](#) di Referensi Kebijakan AWS Terkelola.

AWS kebijakan terkelola: AWSVendorInsightsVendorReadOnly

Anda dapat melampirkan kebijakan AWSVendorInsightsVendorReadOnly ke identitas IAM Anda.

Kebijakan ini memberikan akses hanya-baca untuk melihat profil Wawasan AWS Marketplace Vendor dan sumber daya terkait. Dalam Wawasan AWS Marketplace Vendor, penilai sama dengan pembeli, dan vendor sama dengan penjual untuk keperluan panduan ini.

Untuk melihat izin kebijakan ini, lihat [AWSVendorInsightsVendorReadOnly](#) di Referensi Kebijakan AWS Terkelola.

AWS Marketplace pembaruan kebijakan AWS terkelola

Lihat detail tentang pembaruan kebijakan AWS terkelola AWS Marketplace sejak layanan ini mulai melacak perubahan ini. Untuk peringatan otomatis tentang perubahan pada halaman ini, berlangganan ke umpan RSS pada halaman AWS Marketplace [Riwayat dokumen](#).

Perubahan	Deskripsi	Tanggal
AWSMarketplaceSellerFullAccess — Kebijakan yang diperbarui	AWS Marketplace menambahkan empat SellerSettings izin baru untuk fitur profil pajak tambahan: ListSupplementalTaxRegistrations, PutSupplementalTaxRegistration, DeleteSupplementalTaxRegistration, GetTaxRegistration	Desember 20, 2024
AWSMarketplaceSellerFullAccess — Kebijakan yang diperbarui AWSMarketplaceSellerProductsFullAccess — Kebijakan yang diperbarui AWSMarketplaceSellerFullAccess — Kebijakan yang diperbarui AWSMarketplaceSellerProductsReadOnly — Kebijakan yang diperbarui	AWS Marketplace menghapus ListTasks, DescribeTask, UpdateTasks, dan CompleteTasks izin.	Desember 10, 2024
AWSMarketplaceSellerOfferManagement — Ditambahkan kebijakan baru	AWS Marketplace menambahkan kebijakan baru: AWSMarketplaceSellerOfferManagement	November 18, 2024

Perubahan	Deskripsi	Tanggal
AWSMarketplaceSellerFullAccess — Kebijakan yang diperbarui	AWS Marketplace menambahkan UploadFiles izin. Perubahan ini memungkinkan penjual untuk menggunakan halaman usang di halaman. Portal Manajemen AWS Marketplace	6 November 2024
AWSMarketplaceSellerFullAccess — Kebijakan yang diperbarui	AWS Marketplace menambahkan ListAssessments dan DescribeAssessments izin. Perubahan tersebut memungkinkan SSLv2 pengguna untuk mengakses data penilaian.	Oktober 22, 2024
AWSMarketplaceSellerProductsFullAccess — Kebijakan yang diperbarui	AWS Marketplace menambahkan ListAssessments dan DescribeAssessments izin. Perubahan tersebut memungkinkan SSLv2 pengguna untuk mengakses data penilaian.	Oktober 22, 2024
AWSMarketplaceSellerProductsReadOnly — Kebijakan yang diperbarui	AWS Marketplace menambahkan ListAssessments dan DescribeAssessments izin. Perubahan tersebut memungkinkan SSLv2 pengguna untuk mengakses data penilaian.	Oktober 22, 2024

Perubahan	Deskripsi	Tanggal
AWSMarketplaceSellerFullAccess — Kebijakan yang diperbarui	Memperbarui AWSMarketplaceSellerFullAccess dokumentasi untuk mencerminkan penghapusan tindakan berikut:aws-marketplace-management:viewMarketing ,aws-marketplace-management:viewSettings , danaws-marketplace-management:uploadFiles . Pembaruan ini juga mencakup menghapus bagian Menggunakan izin halus.	Juni 4, 2024
AWSMarketplaceGetEntitlements — Kebijakan yang diperbarui	AWS Marketplace diperbarui i AWSMarketplaceGetEntitlements untuk ditambahkan sid untuk pernyataan kebijakan.	Maret 22, 2024
AWSMarketplaceSellerFullAccess — Kebijakan yang diperbarui	AWS Marketplace diperbarui i AWSMarketplaceSellerFullAccess untuk menambahkan izin untuk membuat peran terkait layanan.	Maret 15, 2024
AWSMarketplaceSellerFullAccess — Kebijakan yang diperbarui	AWS Marketplace diperbarui i AWSMarketplaceSellerFullAccess untuk menambahkan izin untuk mengakses informasi pajak.	Februari 8, 2024

Perubahan	Deskripsi	Tanggal
AWSVendorInsightsVendorFullAccess - Kebijakan yang diperbarui	AWS Marketplace diperbarui di AWSVendorInsightsVendorFullAccess untuk menambahkan izin untuk memperbarui sumber data.	18 Oktober 2023
AWSMarketplaceSellerFullAccess — Kebijakan yang diperbarui	AWS Marketplace diperbarui di AWSMarketplaceSellerFullAccess untuk menambahkan izin untuk berbagi entitas.	1 Juni 2023
AWSMarketplaceSellerFullAccess — Kebijakan yang diperbarui	AWS Marketplace diperbarui di AWSMarketplaceSellerFullAccess untuk menambahkan izin yang terkait dengan verifikasi akun, verifikasi rekening bank, manajemen kasus, dan detail pemberitahuan penjual.	1 Juni 2023
AWSMarketplaceSellerFullAccess — Kebijakan yang diperbarui	AWS Marketplace diperbarui di AWSMarketplaceSellerFullAccess untuk menambahkan izin untuk mengakses dasbor penjual.	Desember 23, 2022
AWSMarketplaceSellerFullAccess , AWSMarketplaceSellerProductsFullAccess , AWSMarketplaceSellerProductsReadOnly — Perbarui ke kebijakan yang ada	AWS Marketplace kebijakan yang diperbarui untuk fitur otorisasi berbasis tag baru.	Desember 9, 2022

Perubahan	Deskripsi	Tanggal
AWS Marketplace diperbarui di AWSVendorInsightsVendorFullAccess	AWS Marketplace diperbarui di <code>AWSMarketplaceSellerProductsFullAccess</code> untuk menambahkan pencarian perjanjian, memperbarui snapshot profil, penandaan vendor, dan memungkinkan akses hanya-baca ke laporan AWS Artifact pihak ketiga (pratinjau).	30 November 2022
AWS Marketplace diperbarui di AWSVendorInsightsVendorReadOnly	AWS Marketplace diperbarui di <code>AWSVendorInsightsVendorReadOnly</code> untuk menambahkan izin ke tag daftar dan memungkinkan akses hanya-baca ke laporan AWS Artifact pihak ketiga (pratinjau).	30 November 2022
AWSVendorInsightsVendorFullAccess dan AWSVendorInsightsVendorReadOnly - Ditambahkan kebijakan baru	AWS Marketplace menambahkan kebijakan untuk fitur baru AWS Marketplace Vendor Insights: <code>AWSMarketplaceSellerProductsFullAccess</code> dan <code>AWSVendorInsightsVendorReadOnly</code> .	26 Juli 2022

Perubahan	Deskripsi	Tanggal
AWSMarketplaceSellerProductsFullAccess dan AWSMarketplaceSellerFullAccess — Kebijakan yang diperbarui	AWS Marketplace kebijakan terbaru untuk fitur baru Wawasan AWS Marketplace Vendor: <code>AWSMarketplaceSellerProductsFullAccess</code> dan <code>AWSMarketplaceSellerFullAccess</code> .	26 Juli 2022
AWSMarketplaceSellerFullAccess dan AWSMarketplaceSellerProductsFullAccess — Perbarui ke kebijakan yang ada	AWS Marketplace memperbaiki kebijakan sehingga <code>iam:PassedToService</code> kondisi hanya diterapkan pada <code>iam:PassRole</code> .	22 November 2021
AWSMarketplaceFullAccess – Pembaruan ke kebijakan yang ada	AWS Marketplace menghapus <code>ec2:DescribeAccountAttributes</code> izin duplikat dari <code>AWSMarketplaceFullAccess</code> kebijakan.	20 Juli 2021
AWS Marketplace mulai melacak perubahan	AWS Marketplace mulai melacak perubahan untuk kebijakan yang AWS dikelola.	20 April 2021

AWS Marketplace Izin akun Layanan Analisis Perdagangan

Gunakan kebijakan izin IAM berikut untuk mendaftar di Layanan Analisis AWS Marketplace Perdagangan.

Untuk petunjuk tentang cara mendaftar, ikuti panduan [orientasi](#).

JSON

```
{
```

```
"Version": "2012-10-17",
"Statement": [
  {
    "Effect": "Allow",
    "Action": [
      "iam:ListRoles",
      "iam:CreateRole",
      "iam:CreatePolicy",
      "iam:AttachRolePolicy",
      "aws-marketplace-management:viewReports"
    ],
    "Resource": "*"
  }
]
```

Gunakan kebijakan izin IAM berikut untuk memungkinkan pengguna membuat permintaan ke Layanan Analisis AWS Marketplace Perdagangan.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "marketplacecommerceanalytics:GenerateDataSet",
      "Resource": "*"
    }
  ]
}
```

Untuk informasi selengkapnya tentang fitur ini, lihat [Mengakses data produk dan pelanggan dengan Layanan Analisis AWS Marketplace Perdagangan](#).

Izin Amazon SQS

Sebagai bagian dari proses publikasi produk SaaS, AWS Marketplace memberi Anda topik Amazon SNS yang dapat Anda gunakan untuk menerima pemberitahuan jika status langganan atau hak

pelanggan berubah. Anda dapat mengonfigurasi satu atau beberapa antrian Amazon SQS ke topik sehingga antrian dapat mengambil tindakan pada notifikasi. Misalnya, jika pelanggan menambahkan lebih banyak penyimpanan ke langganan yang mereka miliki ke produk SaaS Anda, topik Amazon SNS dapat mengirim pesan ke antrian Amazon SQS yang memulai proses untuk secara otomatis meningkatkan kapasitas penyimpanan yang tersedia bagi pelanggan tersebut.

Saat Anda berlangganan antrean Amazon Simple Queue Service (Amazon SQS) ke topik Amazon SNS yang disediakan, izin akan ditambahkan secara otomatis untuk memungkinkan topik mempublikasikan pesan ke antrian. Namun, Anda masih memerlukan kebijakan IAM untuk memberikan akses pengguna API Layanan AWS Marketplace Pengukuran dan Hak ke antrian. Ini dapat diterapkan ke pengguna yang sama jika layanan berjalan dengan kredensi yang sama. Buat kebijakan dengan konten berikut dan lampirkan ke pengguna atau peran Anda.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "sqs:ReceiveMessage", "sqs:DeleteMessage",
        "sqs:GetQueueAttributes", "sqs:GetQueueUrl"
      ],
      "Effect": "Allow",
      "Resource": "arn:aws:sqs:REGION_HERE:XXXXXXXXXXXX:NAME_HERE"
    }
  ]
}
```

Note

ResourceField adalah Nama Sumber Daya Amazon (ARN) dari antrian Amazon SQS Anda.

Untuk informasi selengkapnya tentang pemberitahuan pesan dan antrian untuk produk SaaS Anda, lihat dan [the section called “Berlangganan antrian SQS ke topik SNS”](#) [the section called “Mengakses Layanan AWS Marketplace Pengukuran dan Hak APIs”](#)

AWS Marketplace izin API pengukuran dan hak

Produk perangkat lunak sebagai layanan (SaaS), produk Amazon Machine Image (AMI), dan produk kontainer dapat menggunakan dan. AWS Marketplace Metering Service AWS Marketplace Entitlement Service APIs Setiap jenis memerlukan izin AWS Identity and Access Management (IAM) yang berbeda. Untuk produk atau produk Anda, Anda meteran untuk semua penggunaan, dan pelanggan ditagih AWS berdasarkan catatan pengukuran yang Anda berikan. Untuk mengaktifkan integrasi yang diperlukan untuk menyediakan catatan pengukuran AWS Marketplace Anda, akun layanan yang digunakan integrasi memerlukan kebijakan IAM yang dibatasi untuk mengaktifkan akses. Lampirkan kebijakan untuk jenis produk yang Anda kirim informasi pengukuran ke pengguna atau peran yang Anda gunakan untuk integrasi.

Topik

- [Kebijakan IAM untuk produk SaaS](#)
- [Kebijakan IAM untuk produk AMI](#)
- [Kebijakan IAM untuk produk kontainer](#)

Kebijakan IAM untuk produk SaaS

Dalam kebijakan berikut, izin pertama `aws-marketplace:ResolveCustomer`, diperlukan untuk semua integrasi SaaS. Izin kedua `aws-marketplace:BatchMeterUsage`, diperlukan untuk AWS Marketplace Metering Service API. Izin ketiga `aws-marketplace:GetEntitlements`, diperlukan untuk AWS Marketplace Entitlement Service API.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "aws-marketplace:ResolveCustomer",
        "aws-marketplace:BatchMeterUsage",
        "aws-marketplace:GetEntitlements"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

```
]
}
```

Untuk informasi lebih lanjut tentang produk SaaS, lihat [Produk berbasis SaaS di AWS Marketplace](#)

Kebijakan IAM untuk produk AMI

Gunakan kebijakan IAM berikut untuk produk AMI.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        aws-marketplace:MeterUsage
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

Untuk informasi lebih lanjut tentang produk AMI, lihat [Produk berbasis AMI di AWS Marketplace](#).

Kebijakan IAM untuk produk kontainer

Gunakan kebijakan IAM berikut untuk produk kontainer.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Action": [
        "aws-marketplace:RegisterUsage"
      ],
      "Effect": "Allow",
      "Resource": "*"
    }
  ]
}
```

}

Untuk informasi lebih lanjut tentang produk kontainer, lihat [Produk berbasis kontainer pada AWS Marketplace](#).

Untuk informasi selengkapnya tentang membuat pengguna, lihat [Membuat pengguna Akun AWS di Panduan Pengguna IAM](#). Untuk informasi selengkapnya tentang membuat dan menetapkan kebijakan, lihat [Mengubah izin untuk pengguna IAM](#).

Kebijakan ini memberikan akses ke APIs peran IAM atau pengguna yang Anda lampirkan kebijakan. Untuk informasi selengkapnya tentang cara mengaktifkan asumsi peran oleh akun lain untuk panggilan API ini, lihat [Cara Terbaik Merancang Langganan AWS Marketplace SaaS Anda di Akun AWS Banyak di AWS Partner Network Blog \(APN\)](#).

Menggunakan peran terkait layanan untuk Otorisasi Penjualan Kembali dengan AWS Marketplace

AWS Marketplace menggunakan AWS Identity and Access Management peran [terkait layanan](#) (IAM). Peran terkait layanan adalah jenis unik peran IAM yang ditautkan langsung ke AWS Marketplace. Peran terkait layanan telah ditentukan sebelumnya oleh AWS Marketplace dan mencakup semua izin yang diperlukan layanan untuk memanggil AWS layanan lain atas nama Anda.

Peran terkait layanan membuat pengaturan AWS Marketplace lebih mudah karena Anda tidak perlu menambahkan izin yang diperlukan secara manual. AWS Marketplace mendefinisikan izin peran terkait layanan, dan kecuali ditentukan lain, hanya AWS Marketplace dapat mengambil perannya. Izin yang ditentukan mencakup kebijakan kepercayaan dan kebijakan izin, dan kebijakan izin tersebut tidak dapat dilampirkan ke entitas IAM lainnya.

Anda dapat menghapus peran tertaut layanan hanya setelah menghapus sumber daya terkait terlebih dahulu. Ini melindungi AWS Marketplace sumber daya Anda karena Anda tidak dapat secara tidak sengaja menghapus izin untuk mengakses sumber daya.

Untuk informasi tentang layanan lain yang mendukung peran terkait layanan, silakan lihat [layanan AWS yang bisa digunakan dengan IAM](#) dan carilah layanan yang memiliki opsi Ya di kolom Peran terkait layanan. Pilih Ya dengan sebuah tautan untuk melihat dokumentasi peran terkait layanan untuk layanan tersebut.

Topik

- [Izin peran terkait layanan untuk AWS Marketplace](#)
- [Membuat peran terkait layanan untuk AWS Marketplace](#)
- [Mengedit peran terkait layanan untuk AWS Marketplace](#)
- [Menghapus peran terkait layanan untuk AWS Marketplace](#)
- [Wilayah yang Didukung untuk AWS Marketplace peran terkait layanan](#)

Izin peran terkait layanan untuk AWS Marketplace

AWS Marketplace menggunakan peran terkait layanan bernama `AWSServiceRoleForMarketplaceResaleAuthorization`, yang memungkinkan akses ke AWS layanan dan sumber daya yang digunakan atau dikelola oleh AWS Marketplace untuk Otorisasi Penjualan Kembali.

Peran `AWSServiceRoleForMarketplaceResaleAuthorization` terkait layanan mempercayai layanan berikut untuk mengambil peran:

- `resale-authorization.marketplace.amazonaws.com`

Kebijakan izin peran bernama

`AWSMarketplaceResaleAuthorizationServiceRolePolicy` memungkinkan AWS Marketplace untuk menyelesaikan tindakan berikut pada sumber daya yang ditentukan.

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [{
    "Sid": "AllowResaleAuthorizationShareActionsRAMCreate",
    "Effect": "Allow",
    "Action": [
      "ram:CreateResourceShare"
    ],
    "Resource": [
      "arn:aws:ram:*:*:*"
    ],
    "Condition": {
      "StringEquals": {
        "ram:RequestedResourceType": "aws-marketplace:Entity"
      }
    }
  ]
}
```

```

    },
    "ArnLike": {
      "ram:ResourceArn": "arn:aws:aws-
marketplace:*:*:AWSMarketplace/ResaleAuthorization/*"
    },
    "Null": {
      "ram:Principal": "true"
    }
  }
},
{
  "Sid": "AllowResaleAuthorizationShareActionsRAMAssociate",
  "Effect": "Allow",
  "Action": [
    "ram:AssociateResourceShare"
  ],
  "Resource": [
    "arn:aws:ram:*:*:*"
  ],
  "Condition": {
    "Null": {
      "ram:Principal": "false"
    },
    "StringEquals": {
      "ram:ResourceShareName": "AWSMarketplaceResaleAuthorization"
    }
  }
},
{
  "Sid": "AllowResaleAuthorizationShareActionsRAMAccept",
  "Effect": "Allow",
  "Action": [
    "ram:AcceptResourceShareInvitation"
  ],
  "Resource": [
    "arn:aws:ram:*:*:*"
  ],
  "Condition": {
    "StringEquals": {
      "ram:ResourceShareName": "AWSMarketplaceResaleAuthorization"
    }
  }
},
{

```

```

        "Sid": "AllowResaleAuthorizationShareActionsRAMGet",
        "Effect": "Allow",
        "Action": [
            "ram:GetResourceShareInvitations",
            "ram:GetResourceShareAssociations"
        ],
        "Resource": [
            "arn:aws:ram:*:*:*"
        ]
    },
    {
        "Sid": "AllowResaleAuthorizationShareActionsMarketplace",
        "Effect": "Allow",
        "Action": [
            "aws-marketplace:PutResourcePolicy",
            "aws-marketplace:GetResourcePolicy"
        ],
        "Resource": "arn:aws:aws-marketplace:*:*:AWSMarketplace/ResaleAuthorization/*",
        "Condition": {
            "ForAnyValue:StringEquals": {
                "aws:CalledVia": ["ram.amazonaws.com"]
            }
        }
    },
    {
        "Sid": "AllowResaleAuthorizationShareActionsMarketplaceDescribe",
        "Effect": "Allow",
        "Action": [
            "aws-marketplace:DescribeEntity"
        ],
        "Resource": "arn:aws:aws-marketplace:*:*:AWSMarketplace/ResaleAuthorization/*"
    }
]
}

```

Anda harus mengonfigurasi izin agar pengguna, grup, atau peran Anda membuat, mengedit, atau menghapus peran terkait layanan. Untuk informasi selengkapnya, lihat [Izin peran terkait layanan](#) dalam Panduan Pengguna IAM.

Membuat peran terkait layanan untuk AWS Marketplace

Anda tidak perlu membuat peran terkait layanan secara manual. Saat Anda memilih peran terkait layanan di Portal Manajemen AWS Marketplace, AWS Marketplace buat peran terkait layanan untuk Anda.

Untuk membuat peran terkait layanan

1. Di [Portal Manajemen AWS Marketplace](#), masuk ke akun manajemen dan pilih Pengaturan.
2. Di bagian Pengaturan, pilih tab Peran terkait layanan.
3. Pada halaman Peran terkait layanan, pilih Peran terkait layanan untuk Otorisasi Penjualan Kembali atau Integrasi Otorisasi Penjualan Kembali, lalu pilih Buat peran terkait layanan atau Konfigurasi integrasi.
4. Pada halaman Peran terkait layanan untuk Otorisasi Penjualan Kembali atau Buat Integrasi Otorisasi Penjualan Kembali, tinjau informasi dan konfirmasi dengan memilih Buat peran terkait layanan atau Buat integrasi.

Pesan muncul di halaman Peran terkait layanan, yang menunjukkan bahwa peran terkait layanan Otorisasi Penjualan Kembali berhasil dibuat.

Jika Anda menghapus peran terkait layanan, Anda dapat mengikuti langkah-langkah ini untuk membuatnya kembali.

Mengedit peran terkait layanan untuk AWS Marketplace

AWS Marketplace tidak memungkinkan Anda untuk mengedit peran `AWSServiceRoleForMarketplaceResaleAuthorization` terkait layanan. Setelah membuat peran terkait layanan, Anda tidak dapat mengubah nama peran karena berbagai entitas mungkin merujuk peran tersebut. Namun, Anda dapat mengedit penjelasan peran menggunakan IAM. Untuk informasi selengkapnya, lihat [Mengedit peran terkait layanan](#) dalam Panduan Pengguna IAM.

Menghapus peran terkait layanan untuk AWS Marketplace

Jika Anda tidak lagi memerlukan fitur atau layanan yang memerlukan peran terkait layanan, sebaiknya hapus peran tersebut. Dengan begitu, Anda tidak perlu lagi memantau atau memelihara entitas yang tidak digunakan.

Note

Jika vendor perangkat lunak independen (ISVs) tidak memiliki peran, tidak AWS Resource Access Manager akan secara otomatis membagikan Otorisasi Penjualan Kembali baru dengan mitra saluran yang ditargetkan. Jika mitra saluran tidak memiliki peran, tidak AWS Resource Access Manager akan secara otomatis menerima Otorisasi Penjualan Kembali yang ditargetkan kepada mereka.

Untuk menghapus peran tertaut layanan secara manual menggunakan IAM

Gunakan konsol IAM, the AWS CLI, atau AWS API untuk menghapus peran `AWSServiceRoleForMarketplaceResaleAuthorization` terkait layanan. Untuk informasi selengkapnya, lihat [Menghapus peran terkait layanan](#) dalam Panduan Pengguna IAM.

Wilayah yang Didukung untuk AWS Marketplace peran terkait layanan

AWS Marketplace mendukung penggunaan peran terkait layanan di semua Wilayah tempat layanan tersedia. Untuk informasi selengkapnya, lihat [AWS Wilayah dan titik akhir](#).

Panggilan Logging AWS Marketplace Metering API dengan AWS CloudTrail

AWS Marketplace terintegrasi dengan AWS CloudTrail, layanan yang menyediakan catatan tindakan yang diambil oleh pengguna, peran, atau Layanan AWS dalam AWS Marketplace. CloudTrail menangkap panggilan API untuk AWS Marketplace sebagai acara. Panggilan yang diambil termasuk panggilan dari AWS Marketplace konsol dan panggilan kode ke operasi AWS Marketplace API.

CloudTrail diaktifkan pada Akun AWS saat Anda membuat akun. Ketika aktivitas acara yang didukung terjadi di AWS Marketplace, aktivitas tersebut dicatat dalam suatu CloudTrail peristiwa bersama dengan Layanan AWS peristiwa lain dalam riwayat Peristiwa. Anda dapat melihat, mencari, dan mengunduh kejadian terbaru di akun Anda.

Setiap entri kejadian atau log berisi informasi tentang siapa yang membuat permintaan tersebut. Informasi identitas membantu Anda menentukan hal berikut ini:

- Apakah permintaan dibuat dengan root atau kredensi AWS Identity and Access Management pengguna.

- Apakah permintaan tersebut dibuat dengan kredensial keamanan sementara atau tidak untuk peran atau pengguna gabungan.
- Apakah permintaan tersebut dibuat oleh Layanan AWS lain.

AWS Marketplace mendukung pencatatan BatchMeterUsage operasi sebagai peristiwa di file CloudTrail log.

AWS Marketplace Mengukur contoh entri file log API

Contoh: **BatchMeterUsage**

Contoh berikut menunjukkan entri CloudTrail log yang menunjukkan BatchMeterUsage tindakan dari AWS Marketplace Metering Service Ketika penjual [mengirimkan catatan pengukuran untuk melaporkan penggunaan pelanggan mereka](#) untuk produk perangkat lunak sebagai layanan (SaaS) yang terdaftar di AWS Marketplace, entri CloudTrail log ini dicatat di penjual. Akun AWS

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "EX_PRINCIPAL_ID",
    "arn": "arn:aws:iam::123456789012:user/*****",
    "accountId": "123456789012",
    "accessKeyId": "EXAMPLE_KEY_ID",
    "userName": "*****"
  },
  "eventTime": "2018-04-19T16:32:51Z",
  "eventSource": "metering-marketplace.amazonaws.com",
  "eventName": "BatchMeterUsage",
  "awsRegion": "us-east-1",
  "sourceIPAddress": "*****",
  "userAgent": "Coral/Netty14",
  "requestParameters": {
    "usageRecords": [
      {
        "dimension": "Dimension1",
        "timestamp": "Apr 19, 2018 4:32:50 PM",
        "customerIdentifier": "customer1",
        "customerAWSAccountID": "987654321098",
        "quantity": 1
      }
    ]
  }
}
```

```

    ],
    "productCode": "EXAMPLE_proCode"
  },
  "responseElements": {
    "results": [
      {
        "usageRecord": {
          "dimension": "Dimension1",
          "timestamp": "Apr 19, 2018 4:32:50 PM",
          "customerIdentifier": "customer1",
          "customerAWSAccountID": "987654321098",
          "quantity": 1
        },
        "meteringRecordId": "bEXAMPLE-98f0-4e90-8bd2-bf0EXAMPLE1e",
        "status": "Success"
      }
    ],
    "unprocessedRecords": [ ]
  },
  "requestID": "dEXAMPLE-251d-11e7-8d11-1f3EXAMPLE8b",
  "eventID": "cEXAMPLE-e6c2-465d-b47f-150EXAMPLE97",
  "readOnly": false,
  "eventType": "AwsApiCall",
  "recipientAccountId": "123456789012"
}
]
}

```

Contoh: **RegisterUsage** untuk wadah

Contoh berikut menunjukkan entri CloudTrail log yang menunjukkan RegisterUsage tindakan dari AWS Marketplace Metering Service Ketika produk kontainer dengan harga per jam AWS Marketplace digunakan di pembeli Akun AWS, perangkat lunak dalam kontainer memanggil RegisterUsage pembeli Akun AWS untuk memulai pengukuran per jam untuk tugas Amazon Elastic Container Service (Amazon ECS) atau Amazon Elastic Kubernetes Service (Amazon EKS) pod. Entri CloudTrail log ini masuk ke pembeli Akun AWS.

```

{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "EX_PRINCIPAL_ID:botocore-session-1111111111",

```

```
    "arn": "arn:aws:sts::123456789012:assumed-role/Alice/botocore-
session-1111111111",
    "accountId": "123456789012",
    "accessKeyId": "EXAMPLE_KEY_ID",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::123456789012:role/Alice",
        "accountId": "123456789012",
        "userName": "Alice"
      },
      "webIdFederationData": {
        "federatedProvider": "arn:aws:iam::123456789012:oidc-provider/
oidc.eks.us-east-1.amazonaws.com/id/EXAMPLEFA1C58F08CDB049167EXAMPLE",
        "attributes": {}
      },
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2020-07-23T02:19:34Z"
      }
    }
  },
  "eventTime": "2020-07-23T02:19:46Z",
  "eventSource": "metering-marketplace.amazonaws.com",
  "eventName": "RegisterUsage",
  "awsRegion": "us-east-1",
  "sourceIPAddress": "1.2.3.4",
  "userAgent": "aws-cli/1.18.103 Python/3.8.2 Linux/4.14.181-142.260.amzn2.x86_64
botocore/1.17.26",
  "requestParameters": {
    "productCode": "EXAMPLE_proCode",
    "publicKeyVersion": 1
  },
  "responseElements": {
    "signature": "eyJhbGciOiJIUzI1Ni..."
  },
  "requestID": "dEXAMPLE-251d-11e7-8d11-1f3EXAMPLE8b",
  "eventID": "cEXAMPLE-e6c2-465d-b47f-150EXAMPLE97",
  "eventType": "AwsApiCall",
  "recipientAccountId": "123456789012"
}
```

Contoh: **MeterUsage** untuk wadah di Amazon EKS

Contoh berikut menunjukkan entri CloudTrail log yang menunjukkan MeterUsage tindakan dari wadah AWS Marketplace Metering Service for di Amazon EKS. Ketika produk kontainer dengan [pengukuran khusus](#) dari AWS Marketplace digunakan di pembeli Akun AWS, perangkat lunak dalam kontainer memanggil MeterUsage pembeli Akun AWS untuk melaporkan setiap jam. Entri CloudTrail log ini masuk ke pembeli Akun AWS.

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "EX_PRINCIPAL_ID:botocore-session-1111111111",
    "arn": "arn:aws:sts::123456789012:assumed-role/Alice/botocore-session-1111111111",
    "accountId": "123456789012",
    "accessKeyId": "EXAMPLE_KEY_ID",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::123456789012:role/Alice",
        "accountId": "123456789012",
        "userName": "Alice"
      },
      "webIdFederationData": {
        "federatedProvider": "arn:aws:iam::123456789012:oidc-provider/oidc.eks.us-east-1.amazonaws.com/id/EXAMPLEFA1C58F08CDB049167EXAMPLE",
        "attributes": {}
      },
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2020-07-23T01:03:26Z"
      }
    }
  },
  "eventTime": "2020-07-23T01:38:13Z",
  "eventSource": "metering-marketplace.amazonaws.com",
  "eventName": "MeterUsage",
  "awsRegion": "us-east-1",
  "sourceIPAddress": "1.2.3.4",
  "userAgent": "aws-cli/1.18.103 Python/3.8.2 Linux/4.14.181-142.260.amzn2.x86_64 botocore/1.17.26",
```

```

"requestParameters": {
  "timestamp": "Jul 23, 2020 1:35:44 AM",
  "usageQuantity": 1,
  "usageDimension": "Dimension1",
  "productCode": "EXAMPLE_proCode"
},
"responseElements": {
  "meteringRecordId": "bEXAMPLE-98f0-4e90-8bd2-bf0EXAMPLE1e"
},
"requestID": "dEXAMPLE-251d-11e7-8d11-1f3EXAMPLE8b",
"eventID": "cEXAMPLE-e6c2-465d-b47f-150EXAMPLE97",
"eventType": "AwsApiCall",
"recipientAccountId": "123456789012"
}

```

Contoh: **MeterUsage** pada AMIs

Contoh berikut menunjukkan entri CloudTrail log yang menunjukkan MeterUsage tindakan dari AWS Marketplace Metering Service for Amazon Machine Images (AMIs). Ketika produk AMI dengan pengukuran khusus dari AWS Marketplace digunakan di pembeli Akun AWS, perangkat lunak dari AMI memanggil MeterUsage pembeli Akun AWS untuk melaporkan penggunaan setiap jam. Entri CloudTrail log ini masuk ke pembeli Akun AWS.

```

{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "EX_PRINCIPAL_ID:i-exampled859aa775c",
    "arn": "arn:aws:sts::123456789012:assumed-role/Alice/i-exampled859aa775c",
    "accountId": "123456789012",
    "accessKeyId": "EXAMPLE_KEY_ID",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::123456789012:role/Alice",
        "accountId": "123456789012",
        "userName": "Alice"
      },
      "webIdFederationData": {},
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2020-07-10T23:05:20Z"
      }
    }
  }
}

```

```
    },
    "ec2RoleDelivery": "1.0"
  }
},
"eventTime": "2020-07-10T23:06:42Z",
"eventSource": "metering-marketplace.amazonaws.com",
"eventName": "MeterUsage",
"awsRegion": "us-east-1",
"sourceIPAddress": "1.2.3.4",
"userAgent": "aws-cli/1.16.102 Python/2.7.16 Linux/4.14.133-113.112.amzn2.x86_64
botocore/1.12.92",
"requestParameters": {
  "productCode": "EXAMPLE_proCode",
  "timestamp": "Jul 10, 2020 11:06:41 PM",
  "usageDimension": "Dimension1",
  "usageQuantity": 1,
  "dryRun": false
},
"responseElements": {
  "meteringRecordId": "bEXAMPLE-98f0-4e90-8bd2-bf0EXAMPLE1e"
},
"requestID": "dEXAMPLE-251d-11e7-8d11-1f3EXAMPLE8b",
"eventID": "cEXAMPLE-e6c2-465d-b47f-150EXAMPLE97",
"eventType": "AwsApiCall",
"recipientAccountId": "123456789012"
}
```

Pemecahan Masalah

Pesan kesalahan pendaftaran penjual umum

Tabel berikut menjelaskan pesan kesalahan pendaftaran penjual umum dan solusinya.

Pesan kesalahan	Solusi
Persyaratan kepatuhan Anda tidak lengkap	Untuk semua penjual Jepang, kami diharuskan mengumpulkan informasi tambahan untuk tujuan peraturan sebelum mereka dapat mempublikasikan produk berbayar. Jika Anda seorang penjual yang berbasis di Jepang dan tertarik untuk mendaftar AWS Marketplace, hubungi tim orientasi penjual Jepang di <aws-jp-marketplace@amazon.co.jp> untuk detail tambahan. Untuk informasi selengkapnya, lihat FAQ JP SOR .
AWS Marketplace tidak dapat memproses permintaan Anda sekarang karena masalah internal. Coba lagi nanti. Jika masalah berlanjut, hubungi kami untuk bantuan.	Bersihkan cache browser Anda dan coba lagi. Jika masalah berlanjut, hubungi tim Operasi AWS Marketplace Penjual .
Sebelum Anda dapat memperbarui produk Anda ke publik, Anda perlu menambahkan profil publik ke akun penjual Anda.	Tambahkan profil publik ke akun penjual Anda sebelum memperbarui produk ke publik. Di Portal AWS Marketplace Manajemen , buka Pengaturan, lalu Profil publik.
Perbarui pengaturan pembayaran Anda agar kompatibel dengan CurrencyCode.	Perbarui pengaturan pembayaran agar kompatibel dengan kode mata uang. Pastikan untuk menambahkan USD, karena mata uang lokal tidak didukung untuk harga konsumsi. Untuk informasi selengkapnya, lihat Mata uang yang tersedia .
Informasi PPN/GST diperlukan untuk tetap menjadi penjual berbayar.	Anda memiliki dua opsi:

Pesan kesalahan	Solusi
	<ol style="list-style-type: none">1. Perbarui pengaturan pajak di akun utama. Tinjau dokumenta si warisan pajak untuk memahami efeknya pada semua akun yang ditautkan.2. Hapus akun ini dari organisasi dan jadikan sebagai akun mandiri. Ini berarti Anda harus menambahkan VAT/GST pengaturan secara manual di halaman pengaturan pajak di Penagihan.
Lokasi wawancara pajak tidak sesuai dengan lokasi bisnis	<p>Untuk memeriksa pengaturan lokasi saat ini:</p> <ol style="list-style-type: none">1. Lokasi wawancara pajak:<ul style="list-style-type: none">• Di Portal AWS Marketplace Manajemen, buka Pengaturan, lalu Informasi pajak.• Pilih Perbarui informasi pajak, diikuti oleh Pengambilan Ulang.• Setelah Anda memilih Perbarui profil pajak, gulir ke bawah dan periksa negara yang telah Anda sediakan.2. Lokasi bisnis:<ul style="list-style-type: none">• Di Portal AWS Marketplace Manajemen, periksa Lokasi bisnis yang disediakan dalam Ringkasan akun pada tab Pengaturan. <p>Jika lokasi tidak cocok:</p> <ul style="list-style-type: none">• Memperbarui informasi pajak di Penagihan: Untuk informasi selengkapnya, lihat Lokasi akun• Perbarui wawancara pajak dalam pengaturan AMMP: Ambil kembali wawancara pajak dari informasi Pajak <p>Pastikan kedua lokasi identik untuk menghindari masalah pemrosesan pembayaran.</p>

Pesan kesalahan	Solusi
<p>Untuk mulai menggunakan AWS Marketplace, lengkapi Pendaftaran AWS Akun</p>	<p>Lengkapi Pendaftaran AWS Akun Anda.</p> <p>Untuk informasi lebih lanjut tentang pendaftaran penjual, lihat Mendaftar sebagai AWS Marketplace penjual.</p>
<p>Profil publik Anda sedang ditinjau. Anda tidak dapat menambahkan metode pencairan sampai profil publik Anda diverifikasi. Periksa lagi dalam 2 hari kerja. Saat ini Anda hanya dapat mempublikasikan produk gratis.</p>	<p>Profil publik Anda sedang ditinjau. Anda tidak dapat menambahkan metode pencairan dan menyelesaikan KYC hingga profil publik Anda diverifikasi. Diperlukan dua hingga tiga hari kerja untuk menyetujui. Saat ini Anda hanya dapat mempublikasikan produk gratis. Setelah profil Anda disetujui, Anda akan menerima email ke alamat email root Anda.</p> <p>Jika Anda masih tidak dapat menambahkan informasi pembayaran setelah tiga hari kerja, hubungi tim Operasi AWS Marketplace Penjual.</p>
<p>Lokasi bisnis Anda berada di yurisdiksi yang tidak didukung. Anda tidak memenuhi syarat untuk mempublikasikan produk berbayar.</p>	<p>Lokasi bisnis Anda berada di yurisdiksi yang tidak didukung. Anda tidak memenuhi syarat untuk mempublikasikan produk berbayar. Untuk informasi lebih lanjut tentang yurisdiksi yang memenuhi syarat, lihat Yurisdiksi yang memenuhi syarat untuk produk berbayar.</p>
<p>AWS akun tidak terdaftar sebagai penjual di AWS Marketplace.</p>	<p>Anda tidak terdaftar sebagai penjual di AWS Marketplace. Pendaftaran lengkap dari Portal AWS Marketplace Manajemen.</p>
<p>Penjual harus memiliki profil publik untuk dapat menjadi penjual berbayar.</p>	<p>Profil publik penjual Anda harus dilengkapi dan disetujui. Lengkapi dan periksa status akun Anda di Portal AWS Marketplace Manajemen.</p>

Riwayat dokumen

Tabel berikut menjelaskan dokumentasi untuk rilis Panduan AWS Marketplace Penjual ini.

Untuk pemberitahuan tentang pembaruan dokumentasi ini, Anda dapat berlangganan umpan RSS.

Perubahan	Deskripsi	Tanggal
Menambahkan langkah untuk mengatur preferensi pencairan ke pendaftaran penjual	Lihat https://docs.aws.amazon.com/marketplace/latest/userguide/seller-registration-process.html langkah-langkah pendaftaran penjual.	21 Mei 2025
Adendum Federal ditambahkan ke Kontrak Standar untuk Marketplace	Adendum ini dapat membantu merampingkan pengadaan Pemerintah AS. Lihat Menggunakan kontrak standar AWS Marketplace untuk informasi lebih lanjut.	April 29, 2025
Pembaruan pedoman SaaS	Menambahkan informasi tentang penunjukan “Deployed on AWS”. Lihat https://docs.aws.amazon.com/marketplace/latest/userguide/saas-guidelines.html untuk informasi lebih lanjut.	April 8, 2025
Surat otorisasi	Menambahkan letter-of-authorization template, digunakan untuk memberikan izin kepada seseorang untuk membuka AWS akun atas nama perusahaan. Lihat Mendaftar sebagai AWS	April 8, 2025

	<p>Marketplace penjual untuk informasi lebih lanjut.</p>	
<p>Menambahkan informasi daftar swalayan ke ML</p>	<p>Lihat https://docs.aws.amazon.com/marketplace/latest/userguide/machine-learning-products untuk informasi daftar layanan mandiri.</p>	<p>Maret 26, 2025</p>
<p>Kebijakan terkelola yang diperbarui</p>	<p>Izin berikut ditambahkan ke AWSMarketplaceSellerFullAccess kebijakan: ListSupplementalTaxRegistrations, PutSupplementalTaxRegistration DeleteSupplementalTaxRegistration, dan GetTaxRegistration.</p>	<p>Desember 20, 2024</p>
<p>Tingkatkan AWS Marketplace produk Anda dengan media promosi</p>	<p>Ditambahkan panduan untuk cara menggunakan media promosi.</p>	<p>Desember 17, 2024</p>
<p>Kebijakan terkelola yang diperbarui</p>	<p>Izin ListTasks DescribeTask UpdateTasks, dan CompleteTasks izin telah dihapus dari kebijakan AWS terkelola berikut:</p> <ul style="list-style-type: none"> • AWSMarketplaceSellerFullAccess • AWSMarketplaceSellerProductsFullAccess • AWSMarketplaceSellerProductsReadOnly 	<p>Desember 10, 2024</p>

[Ditambahkan dokumentasi untuk Beli baru dengan AWS fitur](#)

Beli dengan AWS memungkinkan penjual AWS Marketplace untuk menampilkan dan menjual AWS Marketplace produk di situs web mereka sendiri. AWS Marketplace penjual dapat berintegrasi dengan tombol AWS Marketplace call-to-action (CTA) dan APIs untuk membangun penemuan perangkat lunak yang AWS didukung dan pengalaman pengadaan di situs web milik penjual.

Desember 4, 2024

[EC2 Produk berbasis komponen Image Builder](#)

EC2 Produk berbasis komponen Image Builder ditambahkan sebagai metode pengiriman produk.

Desember 1, 2024

[Asisten AWS Mitra Baru](#)

AWS Asisten Mitra adalah chatbot generatif yang didukung AI untuk Mitra. AWS Ini dapat diakses dari Partner Central dan AWS Marketplace Portal (AMMP).

November 25, 2024

[Kebijakan terkelola yang diperbarui](#)

Kebijakan berikut sekarang memiliki `uploadFiles` izin:

November 18, 2024

- [awsmarketplacefullaccess](#)
- [AWSMarketplaceSellerFullAccess](#)

Menambahkan kebijakan terkelola baru	Ditambahkan AWSMarketplaceSellerOfferManagement dokumentasi.	November 18, 2024
Kebijakan terkelola yang diperbarui	Kebijakan AWS terkelola berikut sekarang memiliki ListAssessments izin DescribeAssessment dan: <ul style="list-style-type: none">• AWSMarketplaceSellerFullAccess• AWSMarketplaceSellerProductsFullAccess• AWSMarketplaceSellerProductsReadOnly	Oktober 9, 2024
Izin usang yang dihapus	Diperbarui AWSMarketplaceSellerFullAccess dokumentasi untuk menghapus tiga tindakan.	Juni 4, 2024
Opsi amandemen perjanjian tahunan Amazon Machine Image (AMI) yang diperbarui	Penjual sekarang dapat menambahkan atau mengganti jenis instance untuk langganan yang ada.	30 Mei 2024
Memperbarui proses Know Your Customer (KYC)	Menambahkan langkah tambahan ke proses KYC untuk penjual.	21 Mei 2024
Pengalaman penawaran pribadi yang diperbarui untuk AWS Marketplace penjual	Menambahkan konten untuk pengalaman yang ditingkatkan untuk membuat dan mengelola penawaran pribadi.	Mei 20, 2024

Persyaratan yang Diperbarui untuk produk add-on Amazon EKS	Memperbarui bagian “Mempersiapkan produk kontainer Anda sebagai AWS Marketplace add-on” dan menambahkan “Persyaratan konfigurasi add-on dan praktik terbaik untuk penyedia add-on”.	8 Mei 2024
Izin yang diperbarui untuk penjual AWS Marketplace	Contoh izin yang diperbarui untuk memperbaiki kesalahan sintaks.	April 2, 2024
Harga kontrak SaaS yang diperbarui	Konten yang diperbarui terkait dengan pelaporan kelebihan untuk kontrak SaaS pay-as-go dengan model harga.	April 2, 2024
Opsi demo dan penawaran pribadi baru di AWS Marketplace	AWS Marketplace sekarang mendukung demo dan opsi permintaan penawaran pribadi pada halaman detail produk untuk penjual tertentu.	April 1, 2024
Dasbor baru untuk perpajakan	AWS Marketplace sekarang mendukung dasbor perpajakan yang menyediakan visualisasi dan data halus tentang pajak AS dan internasional untuk transaksi di AWS Marketplace	Maret 29, 2024
Diperbarui AWSMarketplaceGetEntitlements	Ditambahkan sid ke kebijakan AWSMarketplaceGetEntitlements dikelola.	Maret 22, 2024

Prosedur layanan profesional yang diperbarui	Memperbarui visibilitas produk pengeditan dan menghapus prosedur produk layanan profesional.	Maret 19, 2024
Kebijakan akses AMI yang diperbarui	Memperbarui bagian untuk memperjelas kebijakan akses AMI khusus Linux dan mirip UNIX.	Maret 19, 2024
Menambahkan informasi optimasi mesin pencari untuk AWS Marketplace	Menambahkan konten yang terkait dengan optimasi mesin pencari untuk AWS Marketplace.	Maret 19, 2024
Kebijakan terkelola yang diperbarui untuk penjual di AWS Marketplace	Diperbarui AWSMarketplaceSellerFullAccess untuk menambahkan izin yang terkait dengan pembuatan peran terkait layanan.	Maret 15, 2024
Peran terkait layanan baru untuk produk di AWS Marketplace	AWS Marketplace sekarang menyediakan peran terkait layanan yang memungkinkan akses ke AWS layanan dan sumber daya yang digunakan atau dikelola oleh AWS Marketplace untuk Otorisasi Penjualan Kembali.	Maret 15, 2024
Kebijakan terkelola yang diperbarui untuk penjual di AWS Marketplace	Diperbarui AWSMarketplaceSellerFullAccess untuk menambahkan izin yang terkait dengan mengakses informasi pajak.	Februari 8, 2024

EventBridge Acara Amazon untuk laporan keamanan	AWS Marketplace sekarang mendukung EventBridge peristiwa Amazon, sebelumnya disebut Amazon CloudWatch Events, ketika laporan kerentanan keamanan tersedia untuk produk penjual.	Januari 31, 2024
Support untuk Amazon EKS add-on	Menambahkan konten dan prosedur yang terkait dengan penerbitan ke add-on Amazon EKS dari produk AWS Marketplace berbasis kontainer.	Januari 29, 2024
Menambahkan dukungan untuk penjual layanan profesional untuk menjual kembali produk	Penjual sekarang dapat menciptakan peluang penjualan kembali untuk mitra saluran sebagai vendor perangkat lunak independen (ISVs).	Januari 18, 2024
Ketersediaan umum untuk perjanjian bertanggal future di AWS Marketplace	All ISVs and AWS Marketplace Channel Partners dapat menentukan tanggal mulai future sebagai bagian dari penerbitan penawaran pribadi dengan harga dimuka.	Januari 16, 2024
Dasbor baru untuk pengguna	AWS Marketplace sekarang mendukung dasbor penggunaan yang menyediakan visualisasi dan data halus untuk pelanggan yang menggunakan SaaS dan produk berbasis penggunaan server.	10 Januari 2024

Opsi penyebaran Quick Launch baru untuk penjual	Menambahkan konten dan prosedur yang terkait dengan opsi penyebaran Quick Launch baru untuk produk perangkat lunak sebagai layanan (SaaS).	November 29, 2023
Jadwal pembayaran yang fleksibel tersedia untuk penawaran pribadi	Jadwal Pembayaran Fleksibel (FPS) untuk penawaran pribadi sekarang tersedia untuk semua pelanggan di AWS Marketplace	17 November 2023
Menambahkan konten swalayan untuk produk kontainer	Menambahkan konten dan prosedur yang terkait dengan tindakan swalayan untuk produk kontainer.	3 November 2023
Dasbor baru untuk perjanjian dan pembaruan	AWS Marketplace sekarang menyediakan dasbor perjanjian dan pembaruan untuk penjual.	31 Oktober 2023
EventBridge Acara Amazon untuk set perubahan	AWS Marketplace sekarang mendukung EventBridge peristiwa Amazon, sebelumnya disebut Amazon CloudWatch Events, ketika set perubahan selesai dengan status berhasil, gagal, atau dibatalkan.	31 Oktober 2023
Kebijakan terkelola yang diperbarui untuk penjual di Wawasan AWS Marketplace Vendor	AWS Marketplace diperbarui di AWSVendorInsightsVendorFullAccess untuk menambahkan izin untuk memperbarui sumber data.	18 Oktober 2023

<u>Mengakhiri dukungan AWS Marketplace untuk Aplikasi Desktop (AMDA)</u>	AWS Marketplace mengakhiri dukungan untuk AMDA pada 2 Oktober 2023. Semua konten dan prosedur yang terkait dengan AMDA telah dihapus dari panduan.	2 Oktober 2023
<u>Menambahkan konten swalayan untuk produk SaaS</u>	Menambahkan konten dan prosedur yang terkait dengan tindakan swalayan untuk produk perangkat lunak sebagai layanan (SaaS).	12 September 2023
<u>Pencairan harian sekarang tersedia untuk penjual AWS Marketplace</u>	Penjual Portal Manajemen AWS Marketplace sekarang memiliki opsi untuk menerima pencairan harian atau bulanan.	7 September 2023
<u>Support untuk Amazon EventBridge</u>	AWS Marketplace sekarang mendukung EventBridge acara ketika penjual menerima penawaran baru.	September 6, 2023
<u>Pengalaman onboarding swalayan baru untuk AWS Marketplace Vendor Insights</u>	AWS Marketplace Vendor Insights sekarang mendukung pengalaman orientasi layanan mandiri.	17 Agustus 2023
<u>Kebijakan terkelola yang diperbarui untuk penjual di AWS Marketplace</u>	Diperbarui AWSMarketplaceSellerFullAccess untuk menambahkan izin yang terkait dengan entitas berbagi.	1 Juni 2023

Kebijakan terkelola yang diperbarui untuk penjual di AWS Marketplace	Diperbarui AWSMarketplaceSellerFullAccess untuk menambahkan izin yang terkait dengan verifikasi akun, verifikasi rekening bank, manajemen kasus, dan detail pemberitahuan penjual.	1 Juni 2023
Menambahkan konten tentang akses berbutir halus untuk AWS Marketplace	Menambahkan informasi ikhtisar, izin, dan prosedur untuk akses berbutir halus di Portal Manajemen AWS Marketplace	1 Juni 2023
Prosedur tambahan untuk produk AMI	Menambahkan prosedur swalayan untuk produk AMI.	12 Mei 2023
Mengakhiri dukungan untuk Program Pembebasan Pajak Amazon dan Layanan Perhitungan Pajak AS di AWS Marketplace	Program Pembebasan Pajak Amazon dan Layanan Perhitungan Pajak AS telah dihapus dari panduan karena dukungan untuk layanan ini berakhir efektif 6 Maret 2023.	6 Maret 2023
Menambahkan prosedur untuk produk kontainer	Menambahkan prosedur baru untuk membuat perubahan pada pengaturan produk kontainer.	13 Februari 2023

Mengakhiri dukungan untuk Koneksi Dukungan AWS Marketplace Produk	AWS Marketplace Koneksi Dukungan Produk dan berbagi detail kontak pelanggan menggunakan Layanan Analisis Perdagangan tidak lagi didukung mulai 30 November 2022. AWS Marketplace menghapus konten Koneksi Dukungan AWS Marketplace Produk dari panduan.	27 Januari 2023
Perubahan nama mitra konsultasi	AWS Marketplace sekarang mengacu pada mitra konsultasi sebagai mitra saluran. Panduan ini diperbarui untuk mencerminkan perubahan nama saja.	26 Januari 2023
Perubahan nama tim layanan pelanggan	Tim Managed Catalog Operations (MCO) mengubah nama mereka menjadi tim Operasi AWS Marketplace Penjual. Panduan ini diperbarui untuk mencerminkan perubahan nama saja.	Januari 24, 2023
Halaman penawaran pribadi	Pembeli yang diautentikasi sekarang dapat melihat penawaran AWS Marketplace pribadi yang diperluas ke mereka Akun AWS di halaman Penawaran Pribadi.	19 Januari 2023

Ditambahkan topik tentang pengaturan produk SaaS	Menambahkan topik baru dengan prosedur untuk membuat perubahan pada pengaturan produk SaaS.	Januari 6, 2023
Kebijakan terkelola yang diperbarui untuk penjual di AWS Marketplace	AWS Marketplace diperbarui di AWSMarketplaceSellerFullAccess untuk menambahkan izin untuk mengakses dasbor penjual.	Desember 23, 2022
Pemberitahuan email yang diperbarui untuk penjual	Penjual sekarang diberitahu ketika penawaran pribadi dipublikasikan.	22 Desember 2022
Uji coba gratis SaaS untuk langganan sekarang tersedia untuk penjual AWS Marketplace	Penjual sekarang dapat membuat uji coba gratis untuk produk berlangganan.	16 Desember 2022
Perbarui pengalaman penjual untuk daftar swalayan AMI (versi 2) di AWS Marketplace	Penjual sekarang AWS Marketplace dapat membuat daftar layanan mandiri produk Single-AMI. Penjual dapat melakukan pembaruan sendiri tanpa waktu tunggu tambahan untuk diproses.	14 Desember 2022

Memperbarui tiga kebijakan untuk otorisasi berbasis tag	Memperbarui tiga kebijakan (AWSMarketplaceSellerFullAccess ,AWSMarketplaceSellerProductsFullAccess , danAWSMarketplaceSellerProductsReadOnly) untuk fitur otorisasi AWS Marketplace berbasis tag.	Desember 9, 2022
Kebijakan terbaru untuk penjual di Wawasan AWS Marketplace Vendor	Kebijakan terkelola yang diperbarui AWSVendorInsightsAssessorFullAccess dan AWSVendorInsightsVendorReadOnly untuk AWS Marketplace penjual Wawasan Vendor.	30 November 2022
Mengontrol akses penjual di Wawasan AWS Marketplace Vendor	Menambahkan topik baru untuk Wawasan AWS Marketplace Vendor untuk menjelaskan tindakan dan izin yang tersedia bagi penjual.	30 November 2022
Memperbarui empat kebijakan terkelola untuk Wawasan AWS Marketplace Vendor	Kebijakan AWSVendorInsightsVendorFullAccess yang diperbarui dan AWSVendorInsightsVendorReadOnly dikelola untuk Wawasan AWS Marketplace Vendor.	28 November 2022

Penjual dapat mempublikasikan produk add-on di Amazon EKS	Integrasi AWS Marketplace dan Amazon EKS memungkinkan penjual untuk mempresentasikan produk mereka di konsol Amazon EKS untuk pembeli.	28 November 2022
Pengaturan Wawasan AWS Marketplace Vendor yang Diperbarui	Memperbarui prosedur penyiapan untuk AWS Marketplace Vendor Insights.	18 November 2022
Memperbarui dua kebijakan untuk Wawasan AWS Marketplace Vendor	Memperbarui dua kebijakan <code>AWSMarketplaceSellerProductsFullAccess</code> dan <code>AWSMarketplaceSellerFullAccess</code> untuk Wawasan AWS Marketplace Vendor.	26 Juli 2022
Menambahkan dua kebijakan untuk AWS Marketplace Vendor Insights, fitur yang menawarkan penilaian risiko perangkat lunak.	Menambahkan dua kebijakan <code>AWSVendorInsightsVendorFullAccess</code> dan <code>AWSVendorInsightsVendorReadOnly</code> untuk AWS Marketplace Vendor Insights fitur yang menawarkan penilaian risiko perangkat lunak.	26 Juli 2022
AWS Marketplace Vendor Insights adalah fitur baru yang ditambahkan AWS Marketplace	AWS Marketplace Vendor Insights adalah fitur yang menawarkan penilaian risiko perangkat lunak. AWS Marketplace Vendor Insights adalah fitur yang menawarkan penilaian risiko perangkat lunak.	26 Juli 2022

AWS Marketplace Pembaruan izin Layanan Analisis Perdagangan	Layanan Analisis AWS Marketplace Perdagangan memiliki izin IAM tambahan.	21 Juli 2022
Bagian Layanan Umpan Data Pengiriman Penjual ditambahkan	Pembaruan khusus dokumentasi untuk menambahkan bagian Layanan Umpan Data Pengiriman Penjual dan mengatur ulang bagian yang terkait dengan umpan data.	15 Juni 2022
Bagian laporan tambahan ditambahkan	Menambahkan bagian baru untuk laporan tambahan yang AWS Marketplace menyediakan peluncuran fitur terbaru.	14 Juni 2022
Uji coba gratis SaaS untuk kontrak sekarang tersedia untuk penjual AWS Marketplace	Penjual sekarang dapat membuat uji coba gratis tanpa pekerjaan pengembangan tambahan dengan menentukan panjang uji coba gratis, dimensi yang tersedia dalam periode uji coba, dan jumlah kapasitas penggunaan gratis yang diterima pelanggan.	31 Mei 2022
Pemberitahuan email ditambahkan ke transaksi pembeli dan penjual	Fitur baru yang memungkinkan pemberitahuan email kepada pembeli dan penjual memverifikasi penawaran dan perjanjian yang dibuat. AWS Marketplace	23 Mei 2022

Contoh ditambahkan untuk membuat daftar produk pembelajaran mesin Anda	Pembaruan khusus dokumentasi ke bagian pembelajaran mesin untuk menyertakan contoh untuk menunjukkan perbandingan tampilan penjual dan pandangan pembeli saat membuat produk pembelajaran mesin.	22 April 2022
Pembaruan ke bagian pembelajaran mesin	Pembaruan khusus dokumentasi telah dilakukan ke bagian pembelajaran mesin untuk memperjelas prosedur.	15 April 2022
Israel sekarang menjadi yurisdiksi yang memenuhi syarat	Warga di Israel sekarang memenuhi syarat untuk menjadi penjual AWS Marketplace.	13 April 2022
Diperbarui untuk item baru dalam umpan data penawaran	Produsen sekarang akan menerima informasi penawaran untuk penawaran yang dibuat oleh mitra saluran mereka. Ini termasuk penawaran di mana akun yang bersangkutan adalah penjual catatan dan produsen untuk penawaran juga.	29 Maret 2022
Pemberitahuan peluang pengecer	Penjual sekarang memiliki kemampuan untuk menerima pemberitahuan untuk peluang pengecer.	28 Maret 2022

Menambahkan video ke produk layanan profesional	Memperbarui halaman produk layanan profesional dengan video yang berisi detail tentang cara mengelola produk layanan.	Februari 24, 2022
Topik baru tentang penerapan solusi integrasi SaaS tanpa server	Informasi baru ditambahkan untuk mengintegrasikan penyebaran SaaS tanpa server, termasuk tautan AWS ke Mulai Cepat untuk referensi tentang langkah-langkah penerapan.	Februari 15, 2022
Pembaruan minimal untuk persyaratan berbasis Container dan bagian AMI	Pembaruan minimal untuk kebijakan dalam persyaratan berbasis Container dan menghapus informasi yang salah untuk kontrak harga AMI	14 Februari 2022
Pembaruan versi kontainer	Pembaruan khusus dokumentasi untuk memperjelas cara mendorong gambar kontainer dan artefak lainnya ke repositori.	Februari 10, 2022
Perbarui ke contoh ResolveCustomer kode untuk produk SaaS	Contoh ResolveCustomer kode untuk produk SaaS telah diperbarui untuk disertakan. CustomerAWSAccountID	3 Februari, 2022

[Ditambahkan dokumentasi untuk mengintegrasikan AWS License Manager dengan AWS Marketplace untuk produk Containers Anywhere](#)

Pembaruan khusus dokumentasi untuk menambahkan panduan terperinci tentang penambahan harga kontrak ke produk AWS Marketplace for Containers Anywhere Anda dengan mengintegrasikan dengan License Manager.

1 Februari 2022

[Pembaruan pemberitahuan SNS SaaS](#)

Pembaruan khusus dokumentasi untuk mengklarifikasi pesan notifikasi SaaS.

Januari 25, 2022

[Kemampuan bagi penjual untuk bertransaksi dengan pembeli berbasis EMEA melalui Amazon Web Services EMEA SARL](#)

AWS Marketplace Penjual yang memenuhi syarat sekarang dapat bertransaksi dengan pelanggan Akun AWS yang berbasis di negara dan wilayah di Eropa, Timur Tengah, dan Afrika (EMEA) melalui Amazon Web Services EMEA SARL.

7 Januari 2022

[Menambahkan dokumentasi untuk membuat opsi pengiriman untuk produk berbasis kontainer dengan metode pengiriman bagan Helm](#)

Penjual sekarang dapat menyediakan opsi pengiriman dengan metode pengiriman bagan Helm. Pembeli dapat menggunakan opsi pengiriman ini untuk meluncurkan aplikasi berbasis kontainer dengan menginstal bagan Helm yang disediakan penjual di lingkungan peluncuran mereka. Saat menyediakan metode pengiriman bagan Helm, penjual dapat mengaktifkan Peluncuran Cepat untuk pembeli. Quick Launch adalah fitur yang dapat digunakan pembeli untuk membuat cluster Amazon EKS baru dengan cepat dan meluncurkan aplikasi berbasis kontainer di atasnya. AWS CloudFormation

29 November 2021

[Perbarui ke kebijakan yang ada](#)

Kebijakan keamanan untuk AWS Marketplace penjual telah diperbarui.

22 November 2021

[Harga kontrak untuk AMI dan produk berbasis Container](#)

Vendor perangkat lunak independen (ISVs) sekarang dapat mencantumkan produk berbasis AMI baru atau produk berbasis Container dan menawarkan harga kontrak di muka untuk pembeli.

17 November 2021

Penandaan terukur vendor	Pembaruan khusus dokumentasi untuk penandaan terukur vendor, termasuk contoh kode.	11 November 2021
Layanan Pemberitahuan Sederhana Amazon untuk produk AMI atau Kontainer	Vendor perangkat lunak independen (ISVs) dapat menerima pemberitahuan saat pelanggan berlangganan atau berhenti berlangganan produk AMI atau Container melalui Amazon Simple Notification Service.	November 10, 2021
Izin penjual baru	AWS Marketplace menambahkan izin baru untuk mengakses tab Penawaran dan Mitra di Portal Manajemen AWS Marketplace	November 9, 2021
Kemampuan untuk menerapkan titik akhir yang dikonfigurasi untuk Inferensi Asinkron untuk produk pembelajaran mesin	Untuk perangkat lunak pembelajaran mesin yang mengharapkan inferensi muatan lebih tinggi daripada maksimum, atau memerlukan waktu pemrosesan yang melebihi waktu pemrosesan maksimum per pemanggilan, pembeli memiliki kemampuan untuk menerapkan titik akhir yang dikonfigurasi untuk Inferensi Asinkron Amazon AI. SageMaker	November 8, 2021

Kebijakan pengembalian dana dan persetujuan	Pembaruan khusus dokumentasi untuk memperjelas kebijakan pengembalian dana dan memindahkan semua informasi pengembalian dana ke satu lokasi pusat di Panduan Penjual.AWS Marketplace	Agustus 20, 2021
Pilih atau unggah EULA untuk penawaran pribadi mitra konsultasi	Vendor perangkat lunak independen sekarang dapat memilih atau mengunggah Perjanjian Lisensi Pengguna Akhir (EULA) saat membuat peluang penjualan kembali untuk mitra konsultasi.	17 Agustus 2021
Dimensi produk khusus untuk produk kontrak SaaS	Vendor perangkat lunak independen (ISVs) sekarang dapat menyesuaikan dimensi produk kontrak SaaS saat menciptakan peluang penjualan kembali untuk mitra konsultasi.	17 Agustus 2021
AWS Marketplace Program Demonstrasi Lapangan	Pembaruan khusus dokumentasi untuk memperjelas persyaratan produk kumpulan data AWS Data Exchange untuk Program Demonstrasi Lapangan. AWS Marketplace	Agustus 3, 2021
Pembaruan pedoman produk SaaS	Pedoman produk untuk produk SaaS telah diperbarui.	29 Juli 2021

Pembaruan persyaratan produk berbasis kontainer	Persyaratan produk berbasis kontainer telah diperbarui.	29 Juli 2021
Pembaruan kebijakan keamanan AMI	Kebijakan keamanan untuk produk AMI telah diperbarui.	29 Juli 2021
Yurisdiksi yang lebih memenuhi syarat	Berikut ini sekarang memenuhi syarat untuk menjadi penjual di AWS Marketplace: Hong Kong SAR dan Qatar.	23 Juni 2021
Ikhtisar umpan data	Pembaruan khusus dokumentasi untuk memberikan gambaran umum tentang struktur umpan data yang tersedia bagi penjual.	23 Juni 2021
Memperbarui bagian pembelajaran mesin	Pembaruan hanya dokumentasi untuk informasi tentang membuat dan memelihara produk pembelajaran mesin.	27 Mei 2021
Pembaruan layanan mandiri untuk produk kontainer	Penjual sekarang memiliki cara yang lebih sederhana dan lebih cepat untuk memperbaiki produk berbasis kontainer mereka melalui Portal Manajemen AWS Marketplace	17 Desember 2020
Layanan profesional	Penjual sekarang dapat menawarkan layanan profesional kepada AWS Marketplace pembeli. Menambahkan bagian layanan profesional ke dokumentasi.	3 Desember 2020

Pembaruan layanan mandiri untuk produk AMI	Penjual sekarang memiliki cara yang lebih sederhana dan lebih cepat untuk memperbaiki produk berbasis Amazon Machine Image (AMI) mereka melalui Portal Manajemen AWS Marketplace.	23 November 2020
Yurisdiksi yang lebih memenuhi syarat	Berikut ini sekarang memenuhi syarat untuk menjadi penjual di AWS Marketplace: Bahrain, Norwegia, Swiss, dan Uni Emirat Arab (UEA).	17 Juni 2020
Anda dapat menawarkan peningkatan dan pembaruan pada penawaran pribadi yang diterima	Untuk kontrak SaaS dan kontrak SaaS dengan produk konsumsi, Anda dapat menawarkan peningkatan dan pembaruan menggunakan penawaran pribadi pada penawaran pribadi yang diterima sebelumnya.	28 Mei 2020
Informasi lebih lanjut tersedia di umpan data	Informasi lebih lanjut dari laporan dipecah menjadi umpan data yang lebih kecil untuk menyederhanakan pencarian dan analisis data.	21 Mei 2020
Ketentuan lisensi standar sekarang tersedia	Anda dapat menawarkan persyaratan lisensi standar sebagai pengganti kustom EULAs untuk menyederhanakan proses kontrak.	28 April 2020

Australia dan Selandia Baru adalah yurisdiksi yang memenuhi syarat	<p>Berikut ini sekarang memenuhi syarat untuk menjadi penjual di AWS Marketplace: (i) Penduduk tetap dan warga negara Australia (AU) Selandia Baru (NZ) atau (ii) entitas bisnis yang diselenggarakan atau didirikan di salah satu area tersebut.</p>	2 April 2020
Produk kontainer sekarang mendukung pengukuran khusus dan peningkatan harga	<p>Jika Anda ingin menentukan unit harga Anda sendiri dan meteran yang digunakan kepada kami untuk penagihan , integrasikan dengan tindakan Layanan AWS Marketplace meterUsage Pengukuran.</p>	14 November 2019
AWS Marketplace mendukung produk data melalui AWS Data Exchange	<p>Anda sekarang dapat menyediakan produk data di AWS Marketplace.</p>	13 November 2019
Memperkenalkan layanan API AWS Marketplace Katalog	<p>Layanan API AWS Marketplace Katalog menyediakan antarmuka API bagi penjual yang disetujui untuk mengelola produk mereka secara terprogram.</p>	12 November 2019
AWS Marketplace mendukung kontainer per jam berbayar	<p>AWS Marketplace sekarang mendukung kontainer per jam berbayar yang berjalan di Amazon Elastic Kubernetes Service (Amazon EKS).</p>	25 Septbucket 2019

Fungsionalitas produk AMI yang diperbarui	Anda sekarang dapat menerapkan AMIs dan fungsi Lambda bersama-sama menggunakan AWS CloudFormation	11 September 2019
Menambahkan bagian Keamanan	Konten keamanan terkonsolidasi di bawah bagian Keamanan baru.	7 Mei 2019
Kebijakan keamanan AMI yang diperbarui	Memperbarui kebijakan keamanan untuk produk AMI	April 11, 2019
Menambahkan informasi versi ke bagian Machine Learning Products	Menambahkan konten yang menjelaskan kemampuan versi produk untuk produk pembelajaran mesin.	21 Maret 2019
Menambahkan bagian Produk Machine Learning	Menambahkan konten untuk menerbitkan produk pembelajaran mesin	28 November 2018
Menambahkan bagian Produk Berbasis Kontainer	Menambahkan konten untuk menerbitkan produk berbasis kontainer.	27 November 2018
Tautan yang diperbarui untuk mengirimkan permintaan bantuan penjual	Mengubah alamat email ke alamat webform.	22 Oktober 2018
Menambahkan kontrak SaaS dengan konten konsumsi	Merestrukturisasi konten SaaS dan menambahkan konten untuk mendukung rilis kontrak SaaS dengan fitur konsumsi.	18 Oktober 2018

Menambahkan konten tentang jadwal pembayaran fleksibel untuk penawaran pribadi	Menambahkan konten untuk mendukung rilis penjadwal pembayaran fleksibel untuk konten penawaran pribadi.	15 Oktober 2018
Diperbarui konten izin IAM	Menambahkan konten untuk mendukung izin IAM baru untuk akses baca AMMP saja.	9 Oktober 2018
Menambahkan konten tentang penawaran pribadi mitra konsultasi	Menambahkan konten untuk mendukung rilis fitur Penawaran Pribadi Mitra Konsultasi.	9 Oktober 2018
Menambahkan konten tentang build gambar pribadi	Menambahkan konten untuk mendukung rilis Private Image Build untuk AMIs fitur.	13 Agustus 2018
Menambahkan panduan optimisasi mesin pencari untuk penjual.	Ditambahkan panduan untuk penjual yang ingin mengoptimalkan produk mereka untuk pencarian.	3 Juli 2018
Tautan yang diperbarui ke AWS Marketplace logo	Tautan yang diperbarui untuk menunjuk ke AWS Marketplace logo baru.	Juni 12, 2018
Menambahkan panduan penjual	Mengonversi semua panduan penjual PDF ke konten online.	9 Mei 2018

AWS Glosarium

Untuk AWS terminologi terbaru, lihat [AWS glosarium di Referensi](#).Glosarium AWS

Terjemahan disediakan oleh mesin penerjemah. Jika konten terjemahan yang diberikan bertentangan dengan versi bahasa Inggris aslinya, utamakan versi bahasa Inggris.