



Guida per l'utente

Amazon Linux 2023



Amazon Linux 2023: Guida per l'utente

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

I marchi e l'immagine commerciale di Amazon non possono essere utilizzati in relazione a prodotti o servizi che non siano di Amazon, in una qualsiasi modalità che possa causare confusione tra i clienti o in una qualsiasi modalità che denigri o discrediti Amazon. Tutti gli altri marchi non di proprietà di Amazon sono di proprietà dei rispettivi proprietari, che possono o meno essere affiliati, collegati o sponsorizzati da Amazon.

Table of Contents

Cos'è Amazon Linux 2023?	1
Cadenza di rilascio	1
Rilasci principali e secondari	2
Utilizzo dei nuovi rilasci	2
Policy di supporto a lungo termine	2
Denominazione e controllo delle versioni	3
Ottimizzazioni relative a operazioni e prestazioni	4
Relazione con Fedora	5
Personalizzato cloud-init	6
Aggiornamenti e funzionalità di sicurezza	7
Gestione degli aggiornamenti	8
Sicurezza nel cloud	8
SELinux modalità	8
Programma di conformità	8
server SSH predefinito	8
Funzionalità principali di OpenSSL 3	9
Servizio di networking	9
Pacchetti di toolchain principali glibc, gcc, binutils	10
Strumento di gestione dei pacchetti	11
Configurazione del server SSH predefinita	11
Funzionalità obsoleta	13
Pacchetti compat-	13
Funzionalità obsoleta interrotta in, rimossa in AL1 AL2	13
x86 a 32 bit (i686) AMIs	14
aws-apitools-*sostituito da AWS CLI	14
systemdsostituisce in upstart AL2	15
Funzionalità obsoleta e rimossa in AL2 AL2023	15
Pacchetti x86 (i686) a 32 bit	16
aws-apitools-*sostituito da AWS CLI	16
amazon-cloudwatch-agentsostituisce awslogs	17
bzsistema di controllo delle revisioni	17
cgroup v1	17
log4jhotpatch () log4j-cve-2021-44228-hotpatch	18
lsb_release e il pacchetto system-lsb-core	18

mcrypt	18
OpenJDK (7) java-1.7.0-openjdk	19
Python 2.7	19
rsyslog-opensslsostituisce rsyslog-gnutls	19
Servizio di informazione di rete (NIS)/yp	20
Più nomi di dominio in Amazon VPC create-dhcp-options	20
Sun RPC in glibc	20
Impronta digitale della chiave OpenSSH nel registro audit	21
ld.goldLinker	21
ping6	21
ftpPackage	21
Obsoleto nel 023 AL2	23
Supporto per runtime x86 (i686) a 32 bit	24
aspell	24
libdbBerkeley DB ()	24
cron	24
IMDSv1	25
pcreversione 1	25
System V init (sysvinit)	25
I pacchetti EOL sono obsoleti	26
Confronto AL2 e 023 AL2	27
Pacchetti aggiunti, aggiornati e rimossi	28
Supporto per ogni rilascio	28
Modifiche alla denominazione e al controllo delle versioni	28
Ottimizzazioni	28
Origine da diversi upstream	29
Servizio di sistema delle reti	29
Programma di gestione dei pacchetti	29
Uso di cloud-init	29
Supporto per ambiente grafico o desktop	30
Tripletta del compilatore	30
Pacchetti x86 (i686) a 32 bit	30
lsb_release e il pacchetto system-lsb-core	31
EPEL	31
axel- HTTP/FTP cliente	33
brotlie libbrotli - compressione	33

collectd- Demone per la raccolta delle statistiche	34
cpulimit	34
exim- agente di trasferimento della posta	34
fuse3- File system in Userspace (FUSE) v3	34
ganglia- Sistema di monitoraggio distribuito	35
git-lfs- controllo delle versioni di file di grandi dimensioni con Git	35
haveged- una fonte di entropia che utilizza l'algoritmo HAVEGE	35
inotify-tools- strumenti da riga di comando di inotify	35
iperf- Benchmark TCP/UDP delle prestazioni	36
jemalloc- implementazione alternativa malloc	36
libbsd- Libreria di funzioni compatibile con BSD	36
libserf- Libreria client HTTP	37
libzstd- libreria di compressione zstd	37
lighttpdserver web	37
lshell- una shell con restrizioni	37
monit- monitoraggio di processi, file, directory e dispositivi	37
nodejs	38
perl-Config-General	38
python2-lockfile- blocco dei file	39
python2-rsa- Python RSA puro	39
python2-simplejson- Routine JSON per Python 2	39
rkhunter- Rootkit Hunter	40
rssh- una shell limitata da usare con OpenSSH	40
sscg- generatore di certificati SSL autofirmati	40
stress- Test di stress	40
stress-ng- Test di stress	41
tmpwatch- rimuove i file in base all'ora dell'ultimo accesso	41
xmlstarlet- utilità XML da riga di comando	41
Python 2.7 è stato sostituito con Python 3	41
Aggiornamenti di sicurezza	42
SELinux	42
OpenSSL 3	42
IMDSv2	43
Rimozione di hotpatch log4j (log4j-cve-2021-44228-hotpatch)	43
Aggiornamenti deterministici per la stabilità	44
gp3come tipo di volume Amazon EBS predefinito	44

Gerarchia dei gruppi di controllo unificati (cgroup v2)	45
systemd timer sostituiscono cron	45
Toolchain migliorata: gccbinutils, e glibc	45
systemd il diario sostituisce rsyslog	46
Dipendenze dei pacchetti ridotte al minimo	47
Modifiche ai pacchetti per curl e libcurl	47
GNU Privacy Guard (GNUPG)	47
Amazon Corretto come JVM predefinita	48
AWS CLIv2	48
Avvio UEFI preferito e sicuro	48
Modifiche alla configurazione predefinita del server SSH	48
Il kernel cambia da AL2023 AL2	49
IPv4 TTL	49
Modifiche alla configurazione del kernel incentrate sulla sicurezza	49
Altre modifiche apportate alla configurazione del kernel	54
Supporto per i file system del kernel	56
/tmpmodifiche	61
Modifiche all'AMI e all'immagine del contenitore	61
Confronto tra AMI Amazon Linux 2 e AL2 023	62
Confronto tra le AMI minime tra Amazon Linux 2 e AL2 023	95
Confronto tra container Amazon Linux 2 e AL2 023	116
Confronto AL1 e 023 AL2	124
Supporto per ogni rilascio	124
systemd sostituisce upstart come sistema init	125
Python 2.6 e 2.7 sono stati sostituiti con Python 3	125
OpenJDK 8 come JDK più vecchio	125
Modifiche al kernel da AL2023 AL1	125
Applicazione di patch live del kernel	125
Supporto del file system del kernel	125
Modifiche alla configurazione del kernel incentrate sulla sicurezza	128
Altre modifiche apportate alla configurazione del kernel	131
AL1 e confronto tra AMI AL2 023	132
AL1 e confronto tra AMI minimali e AL2 023	166
AL1 e confronto tra AL2 023 Container	186
Requisiti di sistema	195
Requisiti della CPU per l'esecuzione di 023 AL2	195

Requisiti della CPU ARM per AL2 023	195
Requisiti della CPU x86-64 per 023 AL2	196
Requisiti di memoria (RAM) per l'esecuzione di 023 AL2	197
Il kernel Amazon Linux	198
Ciclo di vita del kernel	198
Aggiornamenti del kernel	199
Le versioni del kernel si sovrappongono tra le distribuzioni	199
Gestione CVE	200
Cosa dovresti fare	200
Driver NVIDIA	201
Informazioni sul repository NVIDIA	201
Abilitazione del repository NVIDIA	201
Installazione dei driver NVIDIA	202
Installazione del toolkit CUDA	202
Rimozione del repository NVIDIA	203
Desktop grafico	204
Argomenti correlati	204
Pacchetti supplementari per Amazon Linux	205
Cosa sono i pacchetti supplementari per Amazon Linux (o SPAL)?	205
Vantaggi	205
Support dei pacchetti SPAL	206
Segnalazione di problemi relativi ai pacchetti	206
Argomenti correlati	207
Configurare SPAL su AL2023	207
Prerequisiti	208
Verifica dei prerequisiti	208
Installazione di SPAL sul sistema	208
Installazione dei pacchetti SPAL	210
Scaricamento dei pacchetti sorgente SPAL	210
Installazione dei pacchetti SPAL debuginfo	210
Disinstallazione del repository SPAL dal sistema	211
Argomenti correlati	211
Applicazioni in esecuzione	212
Controllo delle risorse con systemd	212
Controllo delle risorse con esecuzione di comandi <code>systemd-run</code> singoli	212
Controllo delle risorse in un servizio systemd	215

Utilizzo cgroups utilities	219
Utilizzo AL2023 su AWS	222
Iniziare con AWS	222
Iscriviti per un Account AWS	222
Crea un utente con accesso amministrativo	223
Concessione dell'accesso programmatico	224
AL2023 su Amazon EC2	226
Avvio di AL2023 tramite la console Amazon EC2	227
Avvio AL2023 utilizzando il parametro SSM e AWS CLI	228
Avvio dell' AL2023 AMI più recente utilizzando CloudFormation	229
Avvio AL2023 tramite un ID AMI specifico	231
AL2023 Deprecazione e ciclo di vita degli AMI	231
AL2023 Connessione alle istanze	232
Confronto tra AL2023 standard (impostazione predefinita) e minima AMIs	232
AL2023 in contenitori	260
AL2023 Immagine del contenitore di base	261
AL2023 Immagine minima del contenitore	263
Creazione di immagini di contenitori essenziali AL2023	265
AL2023 confronto tra elenchi di pacchetti di immagini dei contenitori	269
AL2023 AMI minima rispetto alle immagini dei container	275
AL2023 su Elastic Beanstalk	292
AL2023 CloudShell	292
AL2023 per gli host di container Amazon ECS	293
Modifiche rilevanti di Amazon ECS da allora AL2	293
AMI ottimizzate per Amazon ECS personalizzate	294
Amazon EFS su AL2 023	294
amazon-efs-utils	295
Montaggio di un file system Amazon EFS	295
Amazon EMR attivo AL2023	296
AL2023 versioni basate su Amazon EMR	296
AL2023 basato su Amazon EMR su EKS	296
AL2023 su AWS Lambda	296
Runtime provided.al2023 Lambda	296
AL2023 runtime basati	297
Esercitazioni	298
Installa LAMP su AL2023	298

Fase 1: preparare il server LAMP	299
Fase 2: verificare il server LAMP	304
Fase 3: proteggere il server di database	306
Fase 4: Installazione (facoltativa) phpMyAdmin	307
Risoluzione dei problemi	310
Argomenti correlati	311
Configura SSL/TLS su AL2023	311
Prerequisiti	313
Fase 1: abilitare TLS nel server	314
Fase 2: ottenere un certificato firmato dalla CA	317
Fase 3: testare e proteggere la configurazione di sicurezza	325
Risoluzione dei problemi	329
Ospita un WordPress blog su AL2023	330
Prerequisiti	331
Installa WordPress	331
Fasi successive	342
Aiuto! Il nome DNS pubblico è cambiato e il blog non è accessibile	343
Transizione da Redis 6 a Valkey su 023 AL2	344
Cronologia del supporto per Redis 6	345
Introduzione a Valkey	345
Piano e tempistica della migrazione	345
Opzioni e passaggi di migrazione	345
Argomenti correlati	349
Installa GNOME su AL2023	349
Prerequisiti	349
Installazione	350
Argomenti correlati	350
Configura VNC su AL2023	350
Prerequisiti	351
Fase 1: Installazione	351
Fase 2: Configurazione	351
Passaggio 3: Connect utilizzando un client VNC	353
(Facoltativo) Avvia il servizio all'avvio	353
(Facoltativo) Disattiva la schermata di blocco inattiva	354
Argomenti correlati	354
Utilizzo di Multi-Gen LRU (MGLRU) sui kernel AL2023	354

Configurazione e ottimizzazione	355
Installa LibreOffice su AL2023	356
Prerequisiti	357
Fase 1: Determinare l'architettura del sistema	357
Fase 2: Scarica il pacchetto LibreOffice RPM	358
Passaggio 3: Estrarre l'archivio	358
Fase 4: Installare i pacchetti RPM	359
Fase 5: Verificare l'installazione	359
Avvio LibreOffice	359
Risoluzione dei problemi	360
AL2023 al di fuori di Amazon EC2	362
Scarica 203 immagini di macchine virtuali AL2	362
Configurazioni supportate	362
Requisiti per KVM	363
VMware Requisiti	365
Requisiti Hyper-V	367
AL2Configurazione VM 023	369
Configurazione basata su NoCloud <code>seed.iso</code>	370
VMwareconfigurazione basata su <code>guestinfo</code>	374
AL2023 - Confronto dell'elenco dei pacchetti per l'immagine AMI e KVM standard	376
AL2023 confronto dell'elenco dei pacchetti per l'immagine AMI e VMware OVA standard	401
AL2023 confronto tra elenchi di pacchetti per l'AMI standard e l'immagine Hyper-V	426
Identificazione delle versioni di Amazon Linux	452
<code>/etc/os-release</code>	452
Differenze principali	453
Tipi di campo	453
Esempi di <code>/etc/os-release</code>	455
Confronto con altre distribuzioni	456
Specifico per Amazon Linux	458
<code>/etc/system-release</code>	459
<code>/etc/image-id</code>	459
Esempi specifici di Amazon Linux	459
Codice di esempio	461
Layout del file system	475
<code>/</code>	475
<code>/boot</code>	476

/boot/efi	476
/etc	476
/home	476
/root	477
/srv	477
/tmp	478
/run	479
/usr	479
/usr/bin	480
/usr/include	480
/usr/lib e /usr/lib64	480
/usr/local	480
/usr/share	480
/var	481
/var/cache	481
/var/lib	481
/var/log	481
/var/spool	482
/var/tmp	482
Aggiornamento AL2023	483
Procedure consigliate per la distribuzione sicura degli aggiornamenti	483
Preparazione per gli aggiornamenti minori	487
Preparazione per gli aggiornamenti principali	487
Ricevi notifiche sui nuovi aggiornamenti	488
Aggiornamenti deterministici tramite repository con versioni	489
Controllo degli aggiornamenti ricevuti dai rilasci principali e secondari	489
Differenze tra aggiornamenti delle versioni principali e secondarie	490
Sapere quando sono disponibili gli aggiornamenti	490
Controlla gli aggiornamenti dei pacchetti disponibili nei AL2023 repository	490
Sostituzione dell'istanza	491
Aggiornamenti deterministici sul posto	492
Gestione degli aggiornamenti	500
Verifica degli aggiornamenti dei pacchetti disponibili	501
Applicazione degli aggiornamenti di sicurezza utilizzando DNF e le versioni del repository ..	505
Riavvio automatico del servizio dopo gli aggiornamenti (di sicurezza)	518
Quando è necessario il riavvio per applicare gli aggiornamenti di sicurezza?	519

Avvio di un'istanza con la versione più recente del repository abilitata	520
Ottenere informazioni di supporto per i pacchetti	521
dnf check-release-update	521
Aggiunta, abilitazione o disabilitazione di nuovi repository	525
Aggiunta di repository con cloud-init	528
Applicazione di patch live del kernel	529
Limitazioni	530
Configurazioni e prerequisiti supportati	530
Utilizzo di Kernel Live Patching	531
Aggiornamenti del kernel	536
Versioni del kernel Linux su AL2023	536
AL2023 Aggiornamento a una versione più recente del kernel	537
AL2023 kernels - Domande frequenti	542
Linguaggi e runtime di programmazione	544
C/C++ e Fortran	544
GCC14	545
Confronto tra le versioni linguistiche standard	546
.NET	548
Migrazione a nuove versioni .NET	548
Installazione di .NET	548
Go	549
AL2023 Funzione Lambda: Go	549
Java	549
Node.js	38
Perl	551
Perl modules	552
PHP	552
Migrazione a nuove versioni PHP	552
Migrazione da PHP 7.x	552
moduli PHP	553
Python	553
Migrazione a nuove versioni Python	554
moduli Python	554
Ruby	554
Rust	556
AL2023 Funzione Lambda: Rust	557

Swift	557
AL2023 funzione: Swift	557
TypeScript	558
AL2023 Utenti e gruppi riservati	563
Elenco di AL2 023 utenti riservati	563
Elenco di AL2 023 gruppi riservati	571
Codec disponibili in 023 AL2	584
Sicurezza e conformità	587
Avvisi di sicurezza	588
Annunci ALAS	588
AHIMÈ FAQs	589
Avvisi ALAS	589
Avvisi e archivi RPM	589
Avviso IDs	590
Data di rilascio dell'avviso e data di aggiornamento dell'avviso	590
Tipi di avvisi	591
Severità consultive	591
Avvisi e pacchetti	592
Avvisi e CVEs	592
Testo consultivo	593
Avvisi su Kernel Live Patch	594
Schema updateinfo.xml	595
Elenco degli avvisi applicabili	595
Aggiornamenti sul posto	599
Applicazione degli aggiornamenti menzionati in un avviso	599
Impostazione SELinux delle modalità per AL2 023	603
SELinux Stato e modalità predefiniti per AL2 023	604
Passaggio alla modalità enforcing	604
Opzione da disabilitare SELinux	606
Abilita la modalità FIPS su 023 AL2	607
Abilita la modalità FIPS in un contenitore AL2 023	609
Scambia i provider FIPS OpenSSL su 023 AL2	611
Rafforzamento del kernel	613
Opzioni di rafforzamento del kernel (indipendenti dall'architettura)	613
Opzioni di rafforzamento del kernel specifiche di x86-64	630
Opzioni di rafforzamento del kernel specifiche di aarch64	633

Firma dei metadati del repository	635
Come funziona la firma dei metadati nel repository	635
Differenza tra <code>rpm_gpgcheck</code> e <code>rpm_gpgcheck</code>	636
Abilitazione della verifica dei metadati del repository	636
Verifica del funzionamento della firma dei metadati del repository	637
Chiavi pubbliche GPG per i repository AL2023	637
Avvio sicuro UEFI attivo AL2023	638
Abilita UEFI Secure Boot su AL2023	639
Registrazione di un'istanza esistente	639
Registrazione di un'immagine dallo snapshot	640
Aggiornamenti di revoca	641
Come funziona UEFI Secure Boot su AL2023	641
Registrazione di chiavi personalizzate	642
.....	dcxlili

Cos'è Amazon Linux 2023?

Amazon Linux 2023 (AL2023) è la nuova generazione di Amazon Linux di Amazon Web Services (AWS). Con AL2 023, puoi sviluppare ed eseguire applicazioni cloud e aziendali in un ambiente di runtime sicuro, stabile e ad alte prestazioni. Inoltre, ottieni un ambiente applicativo che offre supporto a lungo termine con accesso alle ultime innovazioni in Linux. AL2023 viene fornito senza costi aggiuntivi.

AL2023 è il successore di Amazon Linux 2 (AL2). Per informazioni sulle differenze tra AL2 023 e AL2, vedere [Package Confronto AL2 e AL2 023 changes in AL2 023](#).

Argomenti

- [Cadenza di rilascio](#)
- [Denominazione e controllo delle versioni](#)
- [Ottimizzazioni relative a operazioni e prestazioni](#)
- [Relazione con Fedora](#)
- [Personalizzato cloud-init](#)
- [Aggiornamenti e funzionalità di sicurezza](#)
- [Servizio di networking](#)
- [Pacchetti di toolchain principali glibc, gcc, binutils](#)
- [Strumento di gestione dei pacchetti](#)
- [Configurazione del server SSH predefinita](#)

Cadenza di rilascio

Amazon Linux 2023 (AL2023) è stato rilasciato a marzo 2023 e sarà supportato fino al 30 giugno 2029. Esistono due fasi di supporto:

- **Supporto standard:** durante questa fase, la versione riceve aggiornamenti trimestrali delle versioni minori. La fase di supporto standard termina il 30 giugno 2027.
- **Manutenzione:** durante questa fase, la versione riceve solo aggiornamenti di sicurezza e correzioni di bug critici. Questi aggiornamenti vengono pubblicati non appena sono disponibili. La fase di manutenzione termina il 30 giugno 2029.

Rilasci principali e secondari

Con ogni rilascio di Amazon Linux (versione principale, versione secondaria o rilascio di sicurezza), rilasciamo una nuova Amazon Machine Image (AMI) Linux.

- Rilascio di versione principale: include nuove funzionalità e miglioramenti in termini di sicurezza e prestazioni in tutto lo stack. I miglioramenti potrebbero includere importanti modifiche al kernel, alla toolchain, Glib C, OpenSSL e qualsiasi altra libreria e utilità di sistema. Le principali versioni di Amazon Linux si basano in parte sulla versione corrente della distribuzione Fedora Linux upstream. AWS potrebbe aggiungere o sostituire pacchetti specifici da altri upstream non appartenenti a Fedora.
- Rilascio di versione secondaria: aggiornamento trimestrale che include aggiornamenti di sicurezza, correzioni di bug, e nuove funzionalità e pacchetti. Ogni versione secondaria è un elenco cumulativo di aggiornamenti che include correzioni di sicurezza e bug oltre a nuove funzionalità e pacchetti. Queste versioni potrebbero includere i runtime delle lingue più recenti, come PHP. Potrebbero includere anche altri pacchetti software popolari come Ansible e Docker.

Utilizzo dei nuovi rilasci

Gli aggiornamenti vengono forniti tramite una combinazione di nuovi rilasci di Amazon Machine Image (AMI) e dei nuovi repository corrispondenti. Per impostazione predefinita, una nuova AMI e il repository a cui essa punta sono abbinati. Tuttavia, puoi indirizzare EC2 le istanze Amazon in esecuzione verso versioni di repository più recenti nel tempo per applicare gli aggiornamenti sulle istanze in esecuzione. Puoi eseguire l'aggiornamento anche lanciando nuove istanze della versione più recente. AMIs

Policy di supporto a lungo termine

Amazon Linux fornisce aggiornamenti per tutti i tuoi pacchetti e mantiene la compatibilità all'interno di una versione principale per le applicazioni basate su Amazon Linux. Pacchetti principali come glibc biblioteca, OpenSSL, OpenSSH, e il DNF i gestori di pacchetti ricevono supporto per tutta la durata della versione principale AL2 023. I pacchetti che non sono inclusi tra quelli principali sono supportati in base alle rispettive origini upstream specifiche. Puoi visualizzare lo stato e le date di supporto specifici dei singoli pacchetti eseguendo il comando seguente.

```
$ sudo dnf supportinfo --pkg packagename
```

Mediante l'esecuzione del comando riportato di seguito puoi ottenere informazioni su tutti i pacchetti attualmente installati.

```
$ sudo dnf supportinfo --show installed
```

L'elenco completo dei pacchetti principali viene finalizzato durante l'anteprima. Se vuoi visualizzare altri pacchetti inclusi tra quelli principali, non esitare a contattarci. Valutiamo mentre raccogliamo feedback. Il feedback su AL2 023 può essere fornito tramite il tuo AWS rappresentante designato o segnalando un problema nel repository [amazon-linux-2023](https://github.com/amazon-linux-2023) su GitHub

Denominazione e controllo delle versioni

AL2023 fornisce una versione minore ogni tre mesi durante i due anni di supporto standard. Ogni rilascio è identificato tramite un incremento da 0 a N. 0 si riferisce al rilascio principale originale dell'iterazione in questione. Tutti i rilasci si chiameranno Amazon Linux 2023. Quando verrà rilasciata la prossima versione di Amazon Linux, AL2 023 accederà al supporto esteso e riceverà aggiornamenti per aggiornamenti di sicurezza e correzioni di bug critici.

Ad esempio, le versioni minori di AL2 023 hanno il seguente formato:

- 2023.0.20230301
- 2023.1.20230601
- 2023.2.20230901

Le AL2 023 corrispondenti AMIs hanno il seguente formato:

- al2023-ami-2023.0.20230301.0-kernel-6.1-x86_64
- al2023-ami-2023.1.20230601.0-kernel-6.1-x86_64
- al2023-ami-2023.2.20230901.0-kernel-6.1-x86_64

All'interno di una versione secondaria specifica, i rilasci periodici delle AMI avvengono con un timestamp della data di rilascio di ciascuna AMI.

- al2023-ami-2023.0.**20230301**.0-kernel-6.1-x86_64
- al2023-ami-2023.0.**20230410**.0-kernel-6.1-x86_64
- al2023-ami-2023.0.**20230520**.0-kernel-6.1-x86_64

Il metodo consigliato per identificare un'istanza AL2 o AL2 023 inizia con la lettura della stringa Common Platform Enumeration (CPE) da `/etc/system-release-cpe`. Quindi, si suddivide la stringa nei relativi campi. Infine, leggi i valori della piattaforma e della versione.

AL2023 introduce anche nuovi file per l'identificazione della piattaforma:

- `/etc/amazon-linux-release` crea symlink a `/etc/system-release`
- `/etc/amazon-linux-release-cpe` crea symlink a `/etc/system-release-cpe`

Questi due file indicano che un'istanza è Amazon Linux. Non è necessario leggere un file o suddividere la stringa in campi a meno che non si desideri conoscere i valori specifici della piattaforma e della versione.

Ottimizzazioni relative a operazioni e prestazioni

Kernel Amazon Linux 6.1

- AL2023 utilizza i driver più recenti per i dispositivi Elastic Network Adapter (ENA) ed Elastic Fabric Adapter (EFA). AL2023 si concentra sui backport di prestazioni e funzionalità per l'hardware nell'infrastruttura Amazon EC2 .
- L'applicazione di patch live del kernel è disponibile per i tipi di istanza `x86_64` e `aarch64`. In tal modo si riduce la necessità di riavviare frequentemente.
- Tutte le configurazioni di build e runtime del kernel includono molte delle stesse ottimizzazioni prestazionali e operative di AL2

Selezione della toolchain di base e flag di build predefiniti

- AL21 pacchetti 023 sono creati con le ottimizzazioni del compilatore (`-O2`) abilitate per impostazione predefinita `-O2`
- AL21 pacchetti 023 sono compilati richiedendo `x86-64v2` for `x86-64` systems (`-march=x86-64-v2`) e Graviton 2 o versioni successive per (`aarch64 -march=armv8.2-a+crypto -mtune=neoverse-n1`
- AL21 pacchetti 023 sono compilati con la vettorizzazione automatica abilitata (`-ftree-vectorize`
- AL21 pacchetti 023 sono creati con Link Time Optimization (LTO) abilitato.
- AL2023 utilizza le versioni aggiornate di Rust, Clang/LLVM e Go.

Selezione e versioni dei pacchetti

- Alcuni backport per i principali componenti di sistema includono diversi miglioramenti delle prestazioni per l'esecuzione sull' EC2 infrastruttura Amazon, in particolare sulle istanze Graviton.
- AL2023 è integrato con diverse funzionalità. Servizi AWS Ciò include SSM Agent, Amazon Kinesis Agent e. AWS CLI CloudFormation
- AL2023 utilizza Amazon Corretto come Java Development Kit (JDK).
- AL2023 fornisce ai motori di database e agli aggiornamenti di runtime del linguaggio di programmazione alle versioni più recenti man mano che vengono rilasciate dai progetti upstream. I runtime del linguaggio di programmazione con nuove versioni vengono aggiunti quando vengono rilasciati.

Implementazione in un ambiente cloud

- L'AMI AL2 023 di base e le immagini del contenitore vengono aggiornate frequentemente per supportare la sostituzione delle istanze con patch.
- Gli aggiornamenti del kernel sono inclusi negli aggiornamenti delle AL2 AMI 023. Ciò significa che non è necessario utilizzare comandi come `yum update` e `reboot` per aggiornare il kernel.
- Oltre all'AMI AL2 023 standard, sono disponibili anche un'AMI e un'immagine contenitore minimali. Scegli l'AMI minima per eseguire un ambiente con il numero minimo di pacchetti necessari per eseguire il servizio.
- Per impostazione predefinita, AL2 023 AMIs e i contenitori sono bloccati su una versione specifica degli archivi dei pacchetti. Non è previsto alcun aggiornamento automatico all'avvio. Ciò significa che hai sempre il controllo su quando importare eventuali aggiornamenti dei pacchetti. Puoi sempre eseguire i test in un ambiente beta/gamma prima di passare alla produzione. Se c'è un problema, puoi usare il percorso di rollback preconvalidato.

Relazione con Fedora

AL2023 mantiene i propri cicli di vita di rilascio e supporto indipendenti da Fedora. AL2023 fornisce versioni aggiornate del software open source, una più ampia varietà di pacchetti e rilasci frequenti. In questo modo vengono preservati i familiari sistemi operativi basati su RPM.

La versione Generally Available (GA) di AL2 023 non è direttamente paragonabile a nessuna versione specifica di Fedora. La versione AL2 023 GA include componenti di Fedora 34, 35 e 36. Alcuni componenti sono identici a quelli di Fedora e altri sono stati modificati. Altri

componenti assomigliano più da vicino ai componenti di CentOS Stream 9 o sono stati sviluppati indipendentemente. Il kernel Amazon Linux proviene dalle opzioni di supporto a lungo termine disponibili su kernel.org, scelte indipendentemente da Fedora.

Personalizzato cloud-init

Il cloud-init package è un'applicazione open source che avvia le immagini Linux in un ambiente di cloud computing. [Per ulteriori informazioni, consulta la documentazione di cloud-init.](#)

AL2023 contiene una versione personalizzata di cloud-init. Con cloud-init, puoi specificare cosa succede all'istanza al momento dell'avvio.

Quando avvii un'istanza, puoi utilizzare i campi user-data per passare azioni a cloud-init. Ciò significa che puoi utilizzare Amazon Machine Images (AMIs) comuni per molti casi d'uso e configurarle dinamicamente all'avvio di un'istanza. AL2023 utilizza anche cloud-init per configurare l'ec2-useraccount.

AL2023 utilizza il cloud-init azioni in `/etc/cloud/cloud.cfg.d` e `/etc/cloud/cloud.cfg`. Puoi crearne uno tuo cloud-init file di azioni nella `/etc/cloud/cloud.cfg.d` directory. Cloud-init legge tutti i file in questa directory in ordine lessicografico. I file più recenti sovrascrivono i valori dei file meno recenti. Quando cloud-init lancia un'istanza, la cloud-init il pacchetto esegue le seguenti attività di configurazione:

- Impostare la lingua locale predefinita
- Impostare il nome host
- Analizzare e gestire i dati utente
- Generare chiavi SSH private host
- Aggiungere chiavi SSH pubbliche di un utente a `.ssh/authorized_keys` per semplificare le procedure di login e amministrazione
- Preparare i repository per la gestione dei pacchetti
- Gestire le azioni dei pacchetti definite nei dati utente
- Eseguire script utente contenuti nei dati utente
- Montare i volumi di archivio dell'istanza, se applicabile
 - Per impostazione predefinita, il volume di archivio dell'istanza `ephemeral0` viene montato in `/media/ephemeral0`, se è presente e include un file system valido. In caso contrario, non viene montato.

- Per impostazione predefinita, vengono montati i volumi di swap associati all'istanza (per i tipi di istanza `m1.small` e `c1.medium`).
- È possibile sovrascrivere il montaggio predefinito del volume dell'Instance Store con quanto segue cloud-init direttiva:

```
#cloud-config
mounts:
- [ ephemeral0 ]
```

Per un maggiore controllo sui supporti, vedi [Supporti in](#) cloud-init documentazione.

- All'avvio di un'istanza, i volumi dell'archivio dell'istanza che supportano TRIM non vengono formattati. Prima di poter montare i volumi dell'archivio dell'istanza, è necessario partizionare e formattare i relativi volumi.

Per ulteriori informazioni, consulta il [supporto TRIM di Instance Store Volume](#) nella Amazon EC2 User Guide.

- All'avvio delle istanze, è possibile utilizzare il modulo `disk_setup` per partizionare e formattare i volumi dell'archivio dell'istanza.

Per ulteriori informazioni, consulta [Disk Setup](#) nel cloud-init documentazione.

Per informazioni sull'utilizzo cloud-init con SELinux, vedi [Utilizzo cloud-init per abilitare la modalità enforcing](#).

Per informazioni su cloud-init formati di dati utente, vedere [User-Data Formats](#) nel cloud-init documentazione.

Aggiornamenti e funzionalità di sicurezza

AL2023 fornisce molti aggiornamenti e soluzioni di sicurezza.

Argomenti

- [Gestione degli aggiornamenti](#)
- [Sicurezza nel cloud](#)
- [SELinux modalità](#)
- [Programma di conformità](#)

- [server SSH predefinito](#)
- [Funzionalità principali di OpenSSL 3](#)

Gestione degli aggiornamenti

Applica gli aggiornamenti di sicurezza utilizzando DNF e versioni del repository. Per ulteriori informazioni, consulta [Gestisci gli aggiornamenti dei pacchetti e del sistema operativo in AL2023](#).

Sicurezza nel cloud

La sicurezza è una responsabilità condivisa tra te AWS e te. Il [modello di responsabilità condivisa](#) lo descrive come sicurezza del cloud e sicurezza nel cloud. Per ulteriori informazioni, consulta [Sicurezza e conformità in Amazon Linux 2023](#).

SELinux modalità

Per impostazione predefinita, SELinux è abilitata e impostata sulla modalità permissiva in AL2 023. In modalità di autorizzazione, le negazioni di autorizzazione vengono registrate ma non applicate.

Le SELinux politiche definiscono le autorizzazioni per utenti, processi, programmi, file e dispositivi. Con SELinux, puoi scegliere una delle due politiche. Le policy sono mirate o di sicurezza multilivello (MLS).

Per ulteriori informazioni sulle SELinux modalità e sulle politiche, consulta [Impostazione SELinux delle modalità per AL2 023](#) e [il SELinux Project Wiki](#).

Programma di conformità

I revisori indipendenti valutano la sicurezza e la conformità di AL2 023 insieme a molti programmi di AWS conformità.

server SSH predefinito

AL2023 include OpenSSH 8.7. OpenSSH 8.7 disabilita per impostazione predefinita l'algoritmo di scambio delle chiavi `ssh-rsa`. Per ulteriori informazioni, consulta [Configurazione del server SSH predefinita](#).

Funzionalità principali di OpenSSL 3

- Il Certificate Management Protocol (CMP, RFC 4210) include sia il protocollo CRMF (RFC 4211) che il protocollo di trasferimento HTTP (RFC 6712).
- A HTTP oppure HTTPS il client in libcrypto supporta GET e POST azioni, reindirizzamento, semplice e ASN.1- contenuti codificati, proxy e timeout.
- Il EVP_KDF funziona con le funzioni di derivazione chiave.
- Il EVP_MAC API funziona con MACs.
- Kernel Linux TLS supporto.

Per ulteriori informazioni, consulta la [guida per la migrazione a OpenSSL](#).

Servizio di networking

Il progetto `systemd-networkd` open source è disponibile a livello esteso nelle moderne distribuzioni Linux. Il progetto usa un linguaggio di configurazione dichiarativo simile al resto del framework `systemd`. I relativi tipi di file di configurazione principali sono i file `.network` e `.link`.

Il pacchetto `amazon-ec2-net-utils` genera configurazioni specifiche dell'interfaccia nella directory `/run/systemd/network`. Queste configurazioni abilitano entrambe IPv4 IPv6 le interfacce quando sono collegate a un'istanza. Queste configurazioni installano anche regole di instradamento delle policy che aiutano a garantire che il traffico con origine locale venga instradato alla rete attraverso l'interfaccia di rete dell'istanza corrispondente. Queste regole assicurano che il traffico corretto venga instradato attraverso l'Elastic Network Interface (ENI) dagli indirizzi o dai prefissi associati. Per ulteriori informazioni sull'uso di ENI, consulta [Using ENI](#) nella Amazon EC2 User Guide.

Puoi personalizzare questo comportamento di rete inserendo un file di configurazione personalizzato nella directory `/etc/systemd/network` per sovrascrivere le impostazioni di configurazione predefinite contenute in `/run/systemd/network`.

La documentazione di [systemd.network](#) descrive in che modo il servizio `systemd-networkd` determina la configurazione da applicare a un'interfaccia specifica. Genera anche nomi alternativi, noti come `altnames`, affinché le interfacce supportate da ENI riflettano le proprietà di varie risorse. AWS Queste proprietà delle interfacce supportate da ENI sono il campo `ENI ID` e `DeviceIndex` del

collegamento ENI. Puoi fare riferimento a queste interfacce utilizzando le relative proprietà quando si utilizzano vari strumenti, come il comando `ip`.

AL2023 nomi di interfaccia di istanza vengono generati utilizzando lo schema di denominazione degli `slotsystemd`. Per ulteriori informazioni, consulta lo [schema di denominazione systemd.net](#).

Inoltre, per impostazione AL2 predefinita, 023 utilizza l'algoritmo di pianificazione della trasmissione di rete per la gestione `fq_code1` attiva delle code. [Per ulteriori informazioni, vedere panoramica CoDel](#)

Pacchetti di toolchain principali glibc, gcc, binutils

Un sottoinsieme di pacchetti in Amazon Linux sono designati come pacchetti toolchain principali. Come parte principale di AL2023, i pacchetti principali ricevono cinque anni di supporto. Potremmo cambiare la versione di un pacchetto, ma il supporto a lungo termine si applica al pacchetto incluso nel rilascio di Amazon Linux.

Questi tre pacchetti principali forniscono una toolchain di sistema che viene utilizzata per sviluppare la maggior parte del software nella distribuzione Amazon Linux.

Pacchetto	Definizione	Scopo
glibc 2.34	Libreria di sistema C	Utilizzata dalla maggior parte dei programmi binari che forniscono funzioni standard e dall'interfaccia tra i programmi e il kernel.
gcc 11.2	Suite di compilatori gcc	Compila C, C++, Fortran.
binutils 2.35	Assembler e linker più altri strumenti binari	Manipola o ispeziona i programmi binari.

Consigliamo di far seguire gli aggiornamenti applicati a tutte le librerie glibc da un riavvio. Per gli aggiornamenti dei pacchetti che controllano i servizi, può essere sufficiente riavviare i servizi per rendere disponibili gli aggiornamenti. Tuttavia, il riavvio del sistema garantisce il completamento di tutti i precedenti aggiornamenti di librerie e pacchetti.

Strumento di gestione dei pacchetti

Lo strumento di gestione dei pacchetti software predefinito in AL2 023 è DNF. DNF è il successore di YUM, lo strumento di gestione dei pacchetti in AL2.

DNF è simile a YUM nel suo utilizzo. Molti DNF i comandi e le opzioni di comando sono gli stessi di YUM comandi. In un comando dell'interfaccia a riga di comando (CLI), nella maggior parte dei casi `dnf` sostituisce `yum`.

Ad esempio, per i seguenti AL2 yum comandi:

```
$ sudo yum install packagename
$ sudo yum search packagename
$ sudo yum remove packagename
```

In AL2 023, diventano i seguenti comandi:

```
$ sudo dnf install packagename
$ sudo dnf search packagename
$ sudo dnf remove packagename
```

In AL2 023 il yum comando è ancora disponibile, ma come puntatore al comando. `dnf` Quindi, quando il yum comando viene utilizzato nella shell o in uno script, tutti i comandi e le opzioni sono uguali a DNF CLI. Per ulteriori informazioni sulle differenze tra YUM CLI e il DNF CLI, vedi [Modifiche in DNF CLI rispetto a YUM](#).

Per un riferimento completo dei comandi e delle opzioni per il comando `dnf`, consulta la pagina `man dnf`. Per ulteriori informazioni, vedere [DNF Riferimento ai comandi](#).

Configurazione del server SSH predefinita

Se disponi di client SSH risalenti a diversi anni fa, potresti visualizzare un errore quando ti connetti a un'istanza. Se l'errore indica che non è stato trovato alcun tipo di chiave host corrispondente, aggiorna la chiave host SSH per risolvere il problema.

Disabilitazione predefinita delle firme `ssh-rsa`

AL2023 include una configurazione predefinita che disabilita l'algoritmo della chiave `ssh-rsa` host legacy e genera un set ridotto di chiavi host. I client devono supportare `ssh-ed25519` o l'algoritmo delle chiavi host `ecdsa-sha2-nistp256`.

La configurazione predefinita accetta uno qualsiasi di questi algoritmi di scambio delle chiavi:

- `curve25519-sha256`
- `curve25519-sha256@libssh.org`
- `ecdh-sha2-nistp256`
- `ecdh-sha2-nistp384`
- `ecdh-sha2-nistp521`
- `diffie-hellman-group-exchange-sha256`
- `diffie-hellman-group14-sha256`
- `diffie-hellman-group16-sha512`
- `diffie-hellman-group18-sha512`

Per impostazione predefinita, AL2 023 genera ed25519 e ECDSA ospita le chiavi. I client supportano `ssh-ed25519` o l'algoritmo delle chiavi host `ecdsa-sha2-nistp256`. Quando ti connetti tramite SSH a un'istanza, devi usare un client che supporti un algoritmo compatibile, ad esempio `ssh-ed25519` o `ecdsa-sha2-nistp256`. Se devi usare altri tipi di chiave, sostituisci l'elenco delle chiavi generate con un frammento `cloud-config` nei dati utente.

Nell'esempio seguente, `cloud-config` genera una chiave host `rsa` con le chiavi `ecdsa` e `ed25519`.

```
#cloud-config
ssh_genkeytypes:
- ed25519
- ecdsa
- rsa
```

Se usi una coppia di chiavi RSA per l'autenticazione della chiave pubblica, il client SSH deve supportare una firma `rsa-sha2-256` o `rsa-sha2-512`. Se usi un client incompatibile e non riesci a eseguire l'aggiornamento, riabilita il supporto per `ssh-rsa` nell'istanza. Per riattivare `ssh-rsa` il supporto, attiva la policy di crittografia LEGACY del sistema utilizzando i seguenti comandi.

```
$ sudo dnf install crypto-policies-scripts
$ sudo update-crypto-policies --set LEGACY
```

Per ulteriori informazioni sulla gestione delle chiavi host, consulta [Amazon Linux Host keys](#).

Funzionalità obsoleta nel 2023 AL2

Le funzionalità obsolete AL2 e non presenti in 023 sono documentate qui. AL2 Si tratta di funzionalità come caratteristiche e pacchetti presenti in 023 AL2, ma non in AL2 023 e che non verranno aggiunte a 023. AL2 Per ulteriori informazioni sulla durata del supporto della funzionalità AL2, vedere Funzionalità [obsoleta](#) in. AL2

C'è anche una funzionalità in AL2 023 che è obsoleta e verrà rimossa in una versione futura. Questo capitolo descrive cos'è questa funzionalità, quando non è più supportata e quando verrà rimossa da Amazon Linux. Comprendere le funzionalità obsolete ti aiuterà a distribuire AL2 023 e a prepararti per la prossima versione principale di Amazon Linux.

Argomenti

- [Pacchetti compat-](#)
- [Funzionalità obsoleta interrotta in, rimossa in AL1 AL2](#)
- [Funzionalità obsoleta e rimossa in AL2 AL2023](#)
- [Obsoleto nel 023 AL2](#)

Pacchetti **compat-**

Tutti i pacchetti AL2 con il prefisso di `compat-` vengono forniti per garantire la compatibilità binaria con i vecchi binari che non sono ancora stati ricostruiti per le versioni moderne del pacchetto. Ogni nuova versione principale di Amazon Linux non includerà alcun `compat-` pacchetto delle versioni precedenti.

Tutti i `compat-` pacchetti in una versione di Amazon Linux (ad esempio AL2) sono obsoleti e non sono presenti nella versione successiva (ad esempio AL2 023). Consigliamo vivamente di ricostruire il software sulla base delle versioni aggiornate delle librerie.

Funzionalità obsoleta interrotta in, rimossa in AL1 AL2

Questa sezione descrive le funzionalità disponibili e non più disponibili in AL1. AL2

Note

Come parte della fase di supporto alla manutenzione di AL1, alcuni pacchetti avevano una data end-of-life (EOL) precedente alla fine di AL1. Per ulteriori informazioni, vedere [AL1 Package support statements](#).

Note

Alcune AL1 funzionalità sono state interrotte nelle versioni precedenti. Per informazioni, consulta le [note AL1 di rilascio](#).

Argomenti

- [x86 a 32 bit \(i686\) AMIs](#)
- [aws-apitools-*sostituito da AWS CLI](#)
- [systemdsostituisce in upstart AL2](#)

x86 a 32 bit (i686) AMIs

Come parte della [versione 2014.09 di](#), AL1 Amazon Linux ha annunciato che sarebbe stata l'ultima versione a produrre 32 bit. AMIs. Pertanto, a partire dalla [versione 2015.03 di](#), AL1 Amazon Linux non supporta più l'esecuzione del sistema in modalità a 32 bit. AL2 offre un supporto di runtime limitato per file binari a 32 bit su host x86-64 e non fornisce pacchetti di sviluppo per consentire la creazione di nuovi binari a 32 bit. AL2023 non include più pacchetti di spazio utente a 32 bit. Consigliamo agli utenti di completare la transizione al codice a 64 bit prima di migrare a 023. AL2

Se è necessario eseguire file binari a 32 bit su AL2 023, è possibile utilizzare lo spazio utente a 32 bit dall'AL2 interno di un AL2 contenitore eseguito su 023. AL2

aws-apitools-* sostituito da AWS CLI

Prima del rilascio di settembre 2013, AWS rendeva disponibile una serie di utilità da riga di comando, implementate in AWS CLI Java, che consentiva agli utenti di effettuare chiamate EC2 API Amazon. Questi strumenti sono stati interrotti nel 2015 e sono AWS CLI diventati il modo preferito per interagire con Amazon EC2 APIs dalla riga di comando. Il set di utilità da riga di comando include i seguenti `aws-apitools-*` pacchetti.

- `aws-apitools-as`
- `aws-apitools-cfn`
- `aws-apitools-common`
- `aws-apitools-ec2`
- `aws-apitools-elb`
- `aws-apitools-mon`

Il supporto upstream per i `aws-apitools-*` pacchetti è terminato a marzo 2017. Nonostante la mancanza di supporto upstream, Amazon Linux ha continuato a fornire alcune di queste utilità da riga di comando, ad esempio per fornire agli utenti `aws-apitools-ec2` la compatibilità con le versioni precedenti. AWS CLI È uno strumento più robusto e completo rispetto ai `aws-apitools-*` pacchetti in quanto viene mantenuto attivamente e fornisce un mezzo per utilizzarli tutti. AWS APIs

I `aws-apitools-*` pacchetti erano obsoleti a marzo 2017 e non riceveranno ulteriori aggiornamenti. Tutti gli utenti di uno di questi pacchetti devono migrare AWS CLI a. Questi pacchetti non sono presenti in AL2 023.

AL1 ha fornito anche `aws-apitools-rds` i pacchetti `aws-apitools-iam` and, che erano obsoleti e non sono più presenti in AL1 Amazon Linux da allora in poi. AL2

systemdsostituisce in upstart AL2

AL2 è stata la prima versione di Amazon Linux a utilizzare il sistema `systemd` `init`, `upstart` in AL1 sostituzione di. Qualsiasi configurazione `upstart` specifica deve essere modificata come parte della migrazione AL1 da una versione più recente di Amazon Linux. Non è possibile utilizzarlo `systemd` su AL1, quindi il passaggio da `upstart` a `systemd` può essere eseguito solo come parte del passaggio a una versione principale più recente di Amazon Linux come AL2 o AL2 023.

Funzionalità obsoleta e rimossa in AL2 AL2023

Questa sezione descrive le funzionalità disponibili e non più disponibili in AL2. AL2023

Argomenti

- [Pacchetti x86 \(i686\) a 32 bit](#)
- [aws-apitools-*sostituito da AWS CLI](#)
- [awslogsobsoleto a favore dell'agente Amazon Logs unificato CloudWatch](#)

- [bzsistema di controllo delle revisioni](#)
- [cgroup v1](#)
- [log4jhotpatch \(\) log4j-cve-2021-44228-hotpatch](#)
- [lsb_release e il pacchetto system-lsb-core](#)
- [mcrypt](#)
- [OpenJDK \(7\) java-1.7.0-openjdk](#)
- [Python 2.7](#)
- [rsyslog-opensslsostituisce rsyslog-gnutls](#)
- [Servizio di informazione di rete \(NIS\)/yp](#)
- [Più nomi di dominio in Amazon VPC create-dhcp-options](#)
- [Sun RPC in glibc](#)
- [Impronta digitale della chiave OpenSSH nel registro audit](#)
- [ld.goldLinker](#)
- [ping6](#)
- [ftpPackage](#)

Pacchetti x86 (i686) a 32 bit

Come parte della [versione 2014.09 di AL1](#), abbiamo annunciato che sarebbe stata l'ultima versione a produrre 32 bit. AMIs Pertanto, a partire dalla [versione 2015.03 di](#), AL1 Amazon Linux non supporta più l'esecuzione del sistema in modalità a 32 bit. AL2 fornisce un supporto di runtime limitato per file binari a 32 bit su host x86-64 e non fornisce pacchetti di sviluppo per consentire la creazione di nuovi binari a 32 bit. AL2023 non include più pacchetti di spazio utente a 32 bit. Consigliamo ai clienti di completare la transizione al codice a 64 bit.

Se è necessario eseguire file binari a 32 bit su AL2023, è possibile utilizzare lo spazio utente a 32 bit dall' AL2 interno di un AL2 contenitore in esecuzione su. AL2023

aws-apitools-* sostituito da AWS CLI

Prima del rilascio di settembre 2013, AWS rendeva disponibile una serie di utilità da riga di comando, implementate inJava, che consentivano ai clienti di effettuare chiamate API Amazon EC2. AWS CLI Questi strumenti sono stati dichiarati obsoleti nel 2015 e sono AWS CLI diventati il modo preferito

per interagire con Amazon EC2 APIs dalla riga di comando. Ciò include i seguenti pacchetti. `aws-apitools-*`

- `aws-apitools-as`
- `aws-apitools-cfn`
- `aws-apitools-common`
- `aws-apitools-ec2`
- `aws-apitools-elb`
- `aws-apitools-mon`

Il supporto upstream per i `aws-apitools-*` pacchetti è terminato a marzo 2017. Nonostante la mancanza di supporto upstream, Amazon Linux ha continuato a fornire alcune di queste utilità da riga di comando (come `aws-apitools-ec2`) per fornire la retrocompatibilità ai clienti. AWS CLI È uno strumento più robusto e completo rispetto ai `aws-apitools-*` pacchetti in quanto viene mantenuto attivamente e fornisce un mezzo per utilizzarli tutti. AWS APIs

I `aws-apitools-*` pacchetti sono stati dichiarati obsoleti a marzo 2017 e non riceveranno ulteriori aggiornamenti. Tutti gli utenti di uno di questi pacchetti devono migrare AWS CLI a. Questi pacchetti non sono presenti in AL2023.

awslogs obsoleto a favore dell'agente Amazon Logs unificato CloudWatch

Il [awslogs](#) pacchetto è obsoleto AL2 e non è più presente in. AL2023 Viene sostituito dall'[agente Unified CloudWatch Logs](#), disponibile nel pacchetto. `amazon-cloudwatch-agent` Per ulteriori informazioni, consulta la [Amazon CloudWatch Logs User Guide](#).

bzr sistema di controllo delle revisioni

Il sistema di controllo delle revisioni [GNU Bazaar](#) (`bzr`) è fuori produzione AL2 e non è più presente in. AL2023

Si consiglia agli utenti di `bzr` migrare i propri repository su. `git`

cgroup v1

AL2023 passa alla gerarchia dei gruppi di controllo unificati (`cgroup v2`), mentre utilizza `cgroup v1`. AL2 Poiché AL2 non supporta `cgroup v2`, questa migrazione deve essere completata come parte del passaggio a. AL2023

log4jhotpatch () **log4j-cve-2021-44228-hotpatch**

Note

Il `log4j-cve-2021-44228-hotpatch` pacchetto è obsoleto e rimosso in AL2 . AL2023

[In risposta a CVE-2021-44228, Amazon Linux ha rilasciato una versione in pacchetto RPM di Hotpatch per Apache Log4j per e. AL1 AL2](#) Nell'[annuncio dell'aggiunta dell'hotpatch ad Amazon Linux, abbiamo notato che «L'installazione dell'hotpatch non sostituisce l'aggiornamento a una versione log4j che mitiga CVE-2021-44228 o CVE-2021-45046»](#).

L'hotpatch era una mitigazione per consentire il tempo necessario per applicare le patch log4j. [La prima versione di disponibilità generale di risale a 15 mesi dopo CVE-2021-44228, quindi non viene fornita con l'hotpatch \(abilitata o meno AL2023 \)](#). AL2023

Si consiglia ai clienti che utilizzano le proprie versioni di log4j su Amazon Linux di assicurarsi di aver effettuato l'aggiornamento alle versioni non interessate da [CVE-2021-44228](#) o [CVE-2021-45046](#).

lsb_release e il pacchetto **system-lsb-core**

Storicamente, alcuni software richiavano il `lsb_release` comando (fornito nel `system-lsb-core` pacchetto) AL2 per ottenere informazioni sulla distribuzione Linux su cui veniva eseguito. La Linux Standards Base (LSB) ha introdotto questo comando e le distribuzioni Linux lo hanno adottato. Le distribuzioni Linux si sono evolute per utilizzare lo standard più semplice per la memorizzazione di queste informazioni in `/etc/os-release` e altri file correlati.

Lo standard `os-release` viene da `systemd`. Per ulteriori informazioni, consulta la [documentazione di systemd os-release](#).

AL2023 non viene fornito con il `lsb_release` comando e non include il `system-lsb-core` pacchetto. Il software deve completare la transizione allo standard `os-release` per mantenere la compatibilità con Amazon Linux e le altre principali distribuzioni Linux.

mcrypt

La `mcrypt` libreria e PHP l'estensione associata erano obsolete in AL2 e non sono più presenti in AL2023

Upstream PHP [ha reso obsoleta l'mcrypt estensione nella versione PHP 7.1](#), che è stata rilasciata per la prima volta a dicembre 2016 e ha avuto la sua versione finale a ottobre 2019.

L'[ultima versione della mcrypt libreria upstream risale al 2007](#) e non ha effettuato la migrazione dal controllo di cvs revisione [SourceForge richiesta per i nuovi commit nel 2017, con il commit più recente \(e solo per i 3 anni precedenti\) risalente al 2011, che rimuoveva la menzione](#) che il progetto aveva un manutentore.

Si consiglia a tutti gli utenti mcrypt rimanenti di di trasferire il proprio codice suOpenSSL, poiché non mcrypt verrà aggiunto a. AL2023

OpenJDK (7) **java-1.7.0-openjdk**

Note

AL2023 fornisce diverse versioni di [Amazon Corretto](#) per supportare carichi di lavoro Java basati. I pacchetti OpenJDK 7 sono AL2 obsoleti in e non sono più presenti in. AL2023 Il JDK più vecchio disponibile AL2023 è fornito da Corretto 8.

Per ulteriori informazioni su Java su Amazon Linux, consulta [Javanel AL2 2023](#).

Python 2.7

Note

AL2023 Python 2.7 è stato rimosso, quindi tutti i componenti del sistema operativo che richiedono Python sono scritti per funzionare con Python 3. Per continuare a usare una versione di Python fornita e supportata da Amazon Linux, converti il codice di Python 2 in Python 3.

Per ulteriori informazioni su Python su Amazon Linux, consulta. [Pythonnel AL2023](#)

rsyslog-openssl sostituisce **rsyslog-gnutls**

Il rsyslog-gnutls pacchetto è obsoleto in e non è più presente in AL2. AL2023 Il rsyslog-openssl pacchetto dovrebbe essere un sostituto immediato per qualsiasi utilizzo del pacchetto. rsyslog-gnutls

Servizio di informazione di rete (NIS)/yp

Il Network Information Service (NIS), originariamente chiamato Yellow Pages o YP è obsoleto e non è più presente in AL2. AL2023 Sono inclusi i seguenti pacchetti: `ypbind`, `eypserv`, `yp-tools`. Questa funzionalità è stata rimossa in altri pacchetti che si integrano con NIS AL2023.

Più nomi di dominio in Amazon VPC `create-dhcp-options`

In Amazon Linux 2, era possibile passare più nomi di dominio nel `domain-name` parametro a [create-dhcp-options](#), il che avrebbe comportato `/etc/resolv.conf` qualcosa di simile `search foo.example.com bar.example.com`. Il DHCP server Amazon VPC invia l'elenco dei nomi di dominio forniti utilizzando DHCP l'opzione 15, che supporta solo un singolo nome di dominio (vedi [RFC 2132](#) sezione 3.17). Poiché AL2023 viene utilizzato `systemd-networkd` per la configurazione di rete, che segue la RFC, questa funzionalità accidentale non è presente AL2 su AL2023

La [AWS CLI documentazione di Amazon VPC](#) dice quanto segue: «Alcuni sistemi operativi Linux accettano più nomi di dominio separati da spazi. Tuttavia, Windows e altri sistemi operativi Linux considerano il valore come un singolo dominio, il che si traduce in un comportamento imprevisto. Se il tuo set di DHCP opzioni è associato a un Amazon VPC con istanze che eseguono sistemi operativi che trattano il valore come un singolo dominio, specifica un solo nome di dominio. »

Su questi sistemi, ad esempio AL2023, se si specificano due domini utilizzando DHCP l'opzione 15 (che ne consente solo uno), e poiché lo [spazio non è valido nei nomi di dominio](#), ciò comporterà la codifica del carattere di spazio come `032`, con conseguente contenitore. `/etc/resolv.conf`
`search foo.exmple.com032bar.example.com`

Per supportare più nomi di dominio, un DHCP server deve utilizzare l'DHCP opzione 119 (vedere [RFC 3397](#), sezione 2). Consulta la [Amazon VPC User Guide per sapere](#) quando questa funzionalità è supportata dal server Amazon VPC. DHCP

Sun RPC in `glibc`

L'implementazione di Sun RPC in `glibc` è obsoleta e rimossa in AL2 AL2023. Si consiglia ai clienti di passare all'utilizzo della `libtirpc` libreria (disponibile in AL2 and AL2023) se sono necessarie Sun RPC funzionalità. L'adozione consente `libtirpc` inoltre alle applicazioni di supportare IPv6.

[Questa modifica riflette la più ampia adozione da parte della comunità della `glibc` rimozione a monte di questa funzionalità, ad esempio la rimozione delle Sun RPC interfacce da Fedora e una modifica `glibc` simile in Gentoo.](#)

Impronta digitale della chiave OpenSSH nel registro **audit**

Più avanti nel ciclo di vita di AL2, è stata aggiunta una patch al pacchetto OpenSSH per emettere l'impronta digitale della chiave utilizzata per l'autenticazione. Questa funzionalità non è presente in AL2023.

ld.goldLinker

Il `ld.gold` linker è disponibile in AL2 e viene rimosso in AL2023. I clienti che creano software che fa riferimento esplicitamente al gold linker devono migrare al linker regular (`ld.bfd`).

Le [note di rilascio originali di GNU Binutils per la versione 2.44](#) (rilasciata a febbraio 2025) documentano la rimozione di `ld.gold`: «Modificando la nostra prassi precedente, in questa versione l'archivio tar `binutils-2.44.tar` non contiene i sorgenti per il gold linker. Questo perché il gold linker è ora obsoleto e alla fine verrà rimosso a meno che i volontari non si facciano avanti e si offrano di continuare lo sviluppo e la manutenzione».

ping6

In AL2023, l'utilità normale supporta IPv6 nativamente e quella separata non `/bin/ping6` è più necessaria. In AL2023, `/usr/sbin/ping6` è un collegamento simbolico all'eseguibile `/usr/bin/ping`.

Questa modifica segue l'adozione da parte della comunità più ampia di `iputils` versioni più recenti che forniscono questa funzionalità, ad esempio la [IPv6 modifica del Ping](#) in Fedora.

ftpPackage

Il `ftp` pacchetto in non AL2 è più disponibile in Amazon Linux a partire da AL2 023. Questa decisione è stata presa come parte del nostro costante impegno per la sicurezza, la manutenibilità e le moderne pratiche di sviluppo del software. Come parte (o prima) della migrazione a AL2 023, consigliamo di migrare qualsiasi utilizzo del `ftp` pacchetto legacy a una delle sue alternative.

Contesto

Il `ftp` pacchetto legacy non viene mantenuto attivamente a monte da molti anni. L'ultimo aggiornamento significativo del codice sorgente è avvenuto all'inizio degli anni 2000 e l'archivio dei sorgenti originale non è più disponibile. Sebbene alcune distribuzioni Linux abbiano installato patch per le vulnerabilità di sicurezza, la codebase rimane in gran parte non gestita.

Alternative consigliate

AL2023 offre diverse alternative moderne e gestite attivamente per la funzionalità FTP:

`lftp`(disponibile nelle versioni 023 AL2 e 023 AL2)

Un sofisticato programma di trasferimento file che supporta FTP, HTTP, SFTP e altri protocolli. Offre più funzionalità rispetto al `ftp` client tradizionale e viene mantenuto attivamente.

Installa con: `dnf install lftp`

`curl`(disponibile nelle versioni AL2 023 AL2 e 03)

Uno strumento a riga di comando versatile per il trasferimento di dati con URLs, che supporta FTP, FTPS, HTTP, HTTPS e molti altri protocolli.

Disponibile per impostazione predefinita in 023 tramite il pacchetto AL2. `curl-minimal` Per un supporto più esteso dei protocolli, puoi opzionalmente passare a `curl-full` using. `dnf swap curl-minimal curl-full`

`wget`(disponibile nelle versioni AL2 0 AL2 e 023)

Un'utilità da riga di comando non interattiva per scaricare file dal Web, che supporta i protocolli HTTP, HTTPS e FTP.

Installa con: `dnf install wget` (non installato di default in tutte le 023 immagini) AL2

`sftp`(disponibile nelle versioni AL2 0 AL2 e 023)

Un protocollo di trasferimento file sicuro che opera tramite SSH e fornisce trasferimenti di file crittografati.

Disponibile per impostazione predefinita come parte del pacchetto OpenSSH.

Considerazioni sulla migrazione

Se le applicazioni o gli script dipendono dal `ftp` client legacy, prendi in considerazione i seguenti approcci di migrazione:

1. Aggiorna gli script per utilizzare alternative moderne: modifica gli script per utilizzare `lftp`, `curl` `wget`, o `sftp` al posto del client legacy. `ftp`
2. Esamina le dipendenze dei pacchetti: alcune applicazioni possono elencare il `ftp` pacchetto come dipendenza nei metadati del pacchetto, anche se da tempo sono migrate internamente all'utilizzo

di protocolli moderni. In questi casi, l'applicazione potrebbe funzionare correttamente su AL2 023 nonostante la mancanza di elementi nel pacchetto. `/usr/bin/ftp ftp` Esamina i requisiti effettivi dell'applicazione anziché fare affidamento esclusivamente sulle dipendenze dichiarate.

3. Aggiorna le dipendenze delle applicazioni: per le applicazioni che gestisci che dichiarano ancora una dipendenza dal `ftp` pacchetto ma non lo utilizzano effettivamente, aggiorna i metadati del pacchetto per rimuovere questa dipendenza non necessaria.

Considerazioni sulla sicurezza

Il protocollo FTP trasmette i dati, incluse le credenziali di autenticazione, in testo semplice. Per le applicazioni sensibili alla sicurezza, consigliamo vivamente di utilizzare alternative crittografate come SFTP o HTTPS, supportate dagli strumenti alternativi consigliati.

Obsoleto nel 023 AL2

Questa sezione descrive le funzionalità presenti in AL2 023 e che probabilmente verranno rimosse in una versione futura di Amazon Linux. Ogni sezione descriverà in cosa consiste la funzionalità e quando è prevista la sua rimozione da Amazon Linux.

Note

Questa sezione verrà aggiornata nel tempo man mano che l'ecosistema Linux si evolve e le future versioni principali di Amazon Linux si avvicinano al rilascio.

Argomenti

- [Supporto per runtime x86 \(i686\) a 32 bit](#)
- [aspell](#)
- [libdbBerkeley DB \(\)](#)
- [cron](#)
- [IMDSv1](#)
- [pcreversione 1](#)
- [System V init \(sysvinit\)](#)
- [I pacchetti EOL sono obsoleti](#)

Supporto per runtime x86 (i686) a 32 bit

AL2023 mantiene la capacità di eseguire file binari x86 (i686) a 32 bit. È probabile che la prossima versione principale di Amazon Linux non supporterà più l'esecuzione di file binari con spazio utente a 32 bit.

aspell

Sebbene AL2 023 venga fornito con il `aspell` pacchetto, è obsoleto e verrà rimosso nella prossima versione principale di Amazon Linux. Si consiglia ai clienti di migrare a sistemi sostitutivi moderni come `o.hunspell` o `enchant2`.

[La deprecazione di `aspell` in AL2 023 segue il più ampio cambiamento della community, ad esempio la deprecazione in `aspell` Fedora](#)

libdb Berkeley DB ()

AL2023 viene fornito con la versione 5.3.28 della libreria Berkeley DB (). `libdb` Questa è l'ultima versione di Berkeley DB prima che la licenza passasse alla licenza GNU Affero GPLv3 (AGPL), dalla meno restrittiva licenza Sleepycat.

Ci sono pochi pacchetti in AL2 023 che continuano a dipendere da Berkeley DB (`libdb`) e la libreria verrà rimossa nella prossima versione principale di Amazon Linux.

Note

Il gestore di pacchetti `dnf` in AL2 023 mantiene il supporto di sola lettura per un database in formato Berkeley DB (BDB). `rpm` Questo supporto verrà rimosso nella prossima versione principale di Amazon Linux.

[La deprecazione di `libdb` segue l'allontanamento della comunità più ampia da esso, ad esempio la deprecazione in `libdb` Fedora](#)

cron

Il `cronie` pacchetto è stato installato di default sull'AL2 AMI, fornendo supporto per il `crontab` metodo tradizionale di pianificazione delle attività periodiche. In AL2 023, `cronie` è incluso per impostazione predefinita. Pertanto, il supporto per `crontab` non è più fornito per impostazione predefinita.

In AL2 023, è possibile installare facoltativamente il `crone` pacchetto per utilizzare i job `classiccron`. Ti consigliamo di eseguire la migrazione ai timer `systemd` grazie alle funzionalità aggiuntive fornite da `systemd`.

È possibile che una versione futura di Amazon Linux, probabilmente la prossima versione principale, non includa più il supporto per i `cron` lavori classici e completi la transizione ai `systemd` timer. Ti consigliamo di abbandonare l'utilizzo. `cron`

IMDSv1

Per impostazione predefinita, AL2 023 AMIs sono configurati per l'avvio in modalità IMDSv2 -only, disabilitando l'uso di. IMDSv1 È ancora possibile utilizzare AL2 023 con abilitato. IMDSv1 È probabile che una versione futura di Amazon Linux venga applicata solo IMDSv2 -only.

Per ulteriori informazioni sulla configurazione IMDS per AMIs, consulta [Configure the AMI](#) nella Amazon EC2 User Guide.

pcrerevisione 1

Il `pcrerevisione 1` pacchetto legacy è obsoleto e verrà rimosso nella prossima versione principale di Amazon Linux. Il pacchetto `pcrerevisione 2` è il successore. Sebbene le prime versioni di AL2 023 fossero state fornite con un numero limitato di pacchetti integratipcrere, questi pacchetti verranno migrati entro 023. pcrere2 AL2 La libreria obsoleta rimarrà disponibile nella versione 023pcrere. AL2

Note

La versione obsoleta di non pcrere riceverà aggiornamenti di sicurezza per l'intera durata di 023. AL2 Per ulteriori informazioni sul ciclo di vita del pcrere supporto e sulla quantità di tempo in cui il pacchetto riceverà gli aggiornamenti di sicurezza, consulta le istruzioni di supporto del [pacchetto](#). pcrere

[La deprecazione di pcrere in favore di pcrere2 segue il più ampio cambiamento della comunità in questa direzione, ad esempio la deprecazione in. pcrereFedora](#)

System V init (**sysvinit**)

Sebbene AL2 023 mantenga la retrocompatibilità con gli script System V service (init), il `systemd` progetto upstream, come parte della sua [versione v254](#), ha annunciato la [deprecazione del supporto](#)

[per gli script di servizio System V e ha indicato che il supporto](#) verrà rimosso in una versione futura di systemd. Per ulteriori informazioni, consulta [systemd](#).

AL2023 manterrà la retrocompatibilità con gli script System V service (init), ma gli utenti sono incoraggiati a migrare all'utilizzo di file systemd unit nativi per essere preparati alla rimozione del supporto per gli script System V service (init) da Amazon Linux, probabilmente nella prossima versione principale.

I pacchetti EOL sono obsoleti

Ogni pacchetto disponibile in AL2 023 ha una [dichiarazione di supporto](#) associata che copre informazioni specifiche di Amazon Linux. Queste dichiarazioni riguardano il nucleo del sistema operativo e la sua durata, oltre a pacchetti come [the section called “PHP”](#) e [the section called “Python”](#), in cui AL2 023 fornisce più versioni, ognuna delle quali è supportata per la durata del progetto Open Source originale.

In AL2 023 è possibile ottenere informazioni di supporto ai pacchetti utilizzando il gestore dei dnf pacchetti. Per ulteriori informazioni, consulta [Ottenere informazioni di supporto per i pacchetti](#).

Se un pacchetto non è più supportato prima della fine della versione principale di Amazon Linux, si deve presumere che questo pacchetto sia obsoleto e non sarà presente nella prossima versione principale di Amazon Linux.

Per pacchetti come [the section called “PHP”](#) e [the section called “Python”](#), in cui ogni versione principale di Amazon Linux ha fornito più versioni, ognuna con un ciclo di vita del supporto diverso, è probabile che continueranno a essere presenti nelle nuove versioni principali di Amazon Linux, anche se con poca o nessuna sovrapposizione delle versioni principali dei pacchetti. Si consiglia di tenere a mente le tempistiche di supporto dei pacchetti Amazon Linux quando si selezionano le dipendenze.

Confronto AL2 e AL2 023

I seguenti argomenti descrivono le differenze principali tra AL2 e AL2 023.

Per ulteriori informazioni sulle funzionalità obsolete in AL1, e AL2 023 AL2, vedere. [Funzionalità obsoleta nel 2023 AL2](#)

Argomenti

- [Pacchetti aggiunti, aggiornati e rimossi](#)
- [Supporto per ogni rilascio](#)
- [Modifiche alla denominazione e al controllo delle versioni](#)
- [Ottimizzazioni](#)
- [Origine da diversi upstream](#)
- [Servizio di sistema delle reti](#)
- [Programma di gestione dei pacchetti](#)
- [Uso di cloud-init](#)
- [Supporto per ambiente grafico o desktop](#)
- [Tripletta del compilatore](#)
- [Pacchetti x86 \(i686\) a 32 bit](#)
- [lsb_release e il pacchetto system-lsb-core](#)
- [Extra Packages for Enterprise Linux \(EPEL\)](#)
- [Python 2.7 è stato sostituito con Python 3](#)
- [Aggiornamenti di sicurezza](#)
- [Aggiornamenti deterministici per la stabilità](#)
- [gp3come tipo di volume Amazon EBS predefinito](#)
- [Gerarchia dei gruppi di controllo unificati \(cgroup v2\)](#)
- [systemd timer sostituiscono cron](#)
- [Toolchain migliorata: gccbinutils, e glibc](#)
- [systemd il diario sostituisce rsyslog](#)
- [Dipendenze dei pacchetti ridotte al minimo](#)
- [Amazon Corretto come JVM predefinita](#)
- [AWS CLIv2](#)

- [Avvio UEFI preferito e sicuro](#)
- [Modifiche alla configurazione predefinita del server SSH](#)
- [AL2023 modifiche al kernel da AL2](#)
- [/tmp è ora tmpfs](#)
- [Modifiche all'AMI e all'immagine del contenitore](#)
- [Confronto dei pacchetti installati su Amazon Linux 2 e Amazon Linux 2023 AMIs](#)
- [Confronto dei pacchetti installati su Amazon Linux 2 e Amazon Linux 2023 Minimal AMIs](#)
- [Confronto dei pacchetti installati sulle immagini dei container di base Amazon Linux 2 e Amazon Linux 2023](#)

Pacchetti aggiunti, aggiornati e rimossi

AL2023 contiene migliaia di pacchetti software disponibili per l'uso. Per un elenco completo di tutti i pacchetti aggiunti, aggiornati o rimossi nella versione AL2 023 rispetto alle versioni precedenti di Amazon Linux, consulta [Package changes in AL2 023](#).

[Per richiedere l'aggiunta o la modifica di un pacchetto nella versione AL2 023, segnala un problema nel repository amazon-linux-2023 su GitHub](#)

Supporto per ogni rilascio

Per AL2 023, offriamo cinque anni di supporto.

Per ulteriori informazioni, consulta [Cadenza di rilascio](#).

Modifiche alla denominazione e al controllo delle versioni

AL2023 supporta gli stessi meccanismi che AL2 supportano l'identificazione della piattaforma. AL2023 introduce anche nuovi file per l'identificazione della piattaforma.

Per ulteriori informazioni, consulta [Denominazione e controllo delle versioni](#).

Ottimizzazioni

AL2023 ottimizza il tempo di avvio per ridurre il tempo che intercorre tra il lancio dell'istanza e l'esecuzione del carico di lavoro del cliente. Queste ottimizzazioni riguardano la configurazione

del kernel dell' EC2 istanza Amazon, `cloud-init` le configurazioni e le funzionalità integrate nei pacchetti del sistema operativo come `e. kmod systemd`

Per ulteriori informazioni su queste ottimizzazioni, consulta [Ottimizzazioni relative a operazioni e prestazioni](#).

Origine da diversi upstream

AL2023 è basato su RPM e include componenti provenienti da più versioni di Fedora e altre distribuzioni, come CentOS 9 Stream. Il kernel Amazon Linux ha origine dai rilasci di supporto a lungo termine (LTS) direttamente da kernel.org, scelti indipendentemente dalle altre distribuzioni.

Per ulteriori informazioni, consulta [Relazione con Fedora](#).

Servizio di sistema delle reti

Il servizio di `systemd-networkd` sistema gestisce le interfacce di rete in 023. AL2 Questa è una modifica da AL2, che utilizza ISC `dhclient` o `dhc1ient`

Per ulteriori informazioni, consulta [Servizio di networking](#).

Programma di gestione dei pacchetti

Lo strumento di gestione dei pacchetti software predefinito su AL2 023 è DNF. DNF è il successore dello strumento YUM di gestione dei pacchetti di AL2

Per ulteriori informazioni, consulta [Strumento di gestione dei pacchetti](#).

Uso di cloud-init

In AL2 023, `cloud-init` gestisce l'archivio dei pacchetti. Per impostazione predefinita, nelle versioni precedenti di Amazon Linux, `cloud-init` installava gli aggiornamenti di sicurezza. Questa non è l'impostazione predefinita per AL2 023. Le nuove funzionalità di aggiornamento deterministico per l'aggiornamento `releasever` all'avvio descrivono il metodo AL2 023 per abilitare gli aggiornamenti dei pacchetti all'avvio. Per ulteriori informazioni, consultare [Gestisci gli aggiornamenti dei pacchetti e del sistema operativo in AL2023](#) e [Aggiornamenti deterministici per la stabilità](#).

Con AL2 023, puoi usare con `cloud-init` SELinux Per ulteriori informazioni, consulta [Utilizzo cloud-init per abilitare la modalità enforcing](#).

Cloud-init carica il contenuto della configurazione con cloud-init da posizioni remote utilizzando HTTP(S). Nelle versioni precedenti, Amazon Linux non ti avvisa quando le risorse remote non sono disponibili. In AL2 023, le risorse remote non disponibili creano un errore irreversibile e falliscono l'esecuzione. Questa modifica di comportamento da AL2, fornisce un comportamento predefinito «fail-closed» più sicuro.

Per ulteriori informazioni, consulta [Personalizzato cloud-init](#) e la [documentazione di cloud-init](#).

Supporto per ambiente grafico o desktop

AL2023 presenta un ambiente desktop grafico basato su GNOME a partire dalla versione 2023.7, che sostituisce il desktop MATE utilizzato in AL2. Questa versione offre agli utenti un'esperienza desktop diversa pur mantenendo le prestazioni ottimizzate per il cloud di 023. L'ambiente desktop GNOME offre varie opzioni di personalizzazione, funzionalità di integrazione del sistema e un design dell'interfaccia utente distinto, fornendo agli utenti un'alternativa al precedente ambiente desktop MATE. Vedi la [AL2Desktop grafico 023](#) pagina per maggiori dettagli.

Tripletta del compilatore

AL2023 imposta la tripletta del compilatore per GCC e LLVM per indicare che amazon è il fornitore.

Quindi, diventa `aarch64-redhat-linux-gcc aarch64-amazon-linux-gcc`

Questo dovrebbe essere completamente trasparente per la maggior parte degli utenti e potrebbe interessare solo coloro che stanno compilando compilatori su AL2 023.

Pacchetti x86 (i686) a 32 bit

Come parte della [versione 2014.09](#) è stato annunciato che AL1 sarebbe stata l'ultima versione a produrre 32 bit. Pertanto, a partire dalla [versione 2015.03 di](#), AL1 Amazon Linux non supportava più l'esecuzione del sistema in modalità a 32 bit. AL2 offriva un supporto di runtime limitato per i file binari a 32 bit su host x86-64 e non forniva pacchetti di sviluppo per consentire la creazione di nuovi binari a 32 bit. AL2023 non include più alcun pacchetto userspace a 32 bit. Ti consigliamo di completare la transizione al codice a 64 bit.

Se è necessario eseguire file binari a 32 bit su AL2 023, è possibile utilizzare lo spazio utente a 32 bit dall'AL2 interno di un AL2 contenitore eseguito su 023.

lsb_release e il pacchetto system-lsb-core

Storicamente, alcuni software richiamavano il `lsb_release` comando (fornito nel AL2 `system-lsb-core` pacchetto) per ottenere informazioni sulla distribuzione Linux su cui veniva eseguito. La Linux Standards Base (LSB) ha introdotto questo comando e le distribuzioni Linux lo hanno adottato. Le distribuzioni Linux si sono evolute per utilizzare lo standard più semplice per la memorizzazione di queste informazioni in `/etc/os-release` e altri file correlati.

Lo standard `os-release` viene da `systemd`. Per ulteriori informazioni, consulta la [documentazione di systemd os-release](#).

AL2023 non viene fornito con il `lsb_release` comando e non include il pacchetto `system-lsb-core`. Il software deve completare la transizione allo standard `os-release` per mantenere la compatibilità con Amazon Linux e le altre principali distribuzioni Linux.

Extra Packages for Enterprise Linux (EPEL)

Warning

L'AL2 `epe1Extra` ha abilitato l'archivio di terze parti EPEL7. A partire dal 30/06/2020, l'EPEL7 archivio di terze parti non viene più mantenuto.

Questo repository di terze parti non avrà aggiornamenti futuri. Ciò significa che non ci saranno correzioni di sicurezza per i pacchetti nell'archivio EPEL.

Questa sezione tratterà le opzioni in AL2 023 per i pacchetti trovati in EPEL.

Extra Packages for Enterprise Linux (EPEL) è un progetto della community Fedora che ha l'obiettivo di creare un ampio array di pacchetti per sistemi operativi Linux di livello enterprise. Il progetto ha prodotto principalmente CentOS pacchetti RHEL e pacchetti. AL2 presenta un elevato livello di compatibilità con CentOS 7. Di conseguenza, molti EPEL7 pacchetti funzionano su AL2.

Non esistono EPEL versioni binarie compatibili con AL2 023. Tuttavia, i clienti che desiderano utilizzare i propri EPEL7 pacchetti in AL2 023 hanno alcune opzioni. Alcuni EPEL pacchetti hanno alternative in AL2 023, mentre altri sono forniti come parte di [Pacchetti supplementari per Amazon Linux](#)

Warning

Aggiungi solo repository progettati per essere usati con AL2 023.

Sebbene i repository progettati per altre distribuzioni possano funzionare oggi, non c'è alcuna garanzia che continueranno a farlo con qualsiasi aggiornamento di pacchetto in AL2 023 o con il repository non progettato per l'uso con 023. AL2

Questa pagina fornisce informazioni sui EPEL7 pacchetti utilizzati dai clienti e sulle loro controparti 023. AL2 AL2

Per il resto dei pacchetti, i clienti potrebbero essere in grado di utilizzare Pacchetti supplementari per Amazon Linux (SPAL). SPAL fornisce migliaia di EPEL9 pacchetti, creati specificamente per Amazon Linux 2023, ma questi pacchetti non sono coperti dai piani di AWS supporto. Ciò significa che CVEs i pacchetti SPAL non vengono tracciati e le patch vengono fornite solo quando sono disponibili a monte.

 Important

Consultate la documentazione prima di utilizzarlo. [Pacchetti supplementari per Amazon Linux](#)

Argomenti

- [axel- HTTP/FTP cliente](#)
- [brotlie libbrotli - compressione](#)
- [collectd- Demone per la raccolta delle statistiche](#)
- [cpulimit- Limitatore di utilizzo della CPU](#)
- [exim- agente di trasferimento della posta](#)
- [fuse3- File system in Userspace \(FUSE\) v3](#)
- [ganglia- Sistema di monitoraggio distribuito](#)
- [git-lfs- controllo delle versioni di file di grandi dimensioni con Git](#)
- [haveged- una fonte di entropia che utilizza l'algoritmo HAVEGE](#)
- [inotify-tools- strumenti da riga di comando di inotify](#)
- [iperf- Benchmark TCP/UDP delle prestazioni](#)
- [jemalloc- implementazione alternativa malloc](#)
- [libbsd- Libreria di funzioni compatibile con BSD](#)
- [libserf- Libreria client HTTP](#)
- [libzstd- libreria di compressione zstd](#)

- [lighttpdserver web](#)
- [lshell- una shell con restrizioni](#)
- [monit- monitoraggio di processi, file, directory e dispositivi](#)
- [nodejs](#)
- [perl-Config-General](#)
- [python2-lockfile- blocco dei file](#)
- [python2-rsa- Python RSA puro](#)
- [python2-simplejson- Routine JSON per Python 2](#)
- [rkhunter- Rootkit Hunter](#)
- [rssh- una shell limitata da usare con OpenSSH](#)
- [sscg- generatore di certificati SSL autofirmati](#)
- [stress- Test di stress](#)
- [stress-ng- Test di stress](#)
- [tmpwatch- rimuove i file in base all'ora dell'ultimo accesso](#)
- [xmlstarlet- utilità XML da riga di comando](#)

axel- HTTP/FTP cliente

Il axel pacchetto era incluso e non è mai stato spedito come parte di Amazon Linux. EPEL7 Le alternative disponibili in AL2 023 sono curl 2 e 3. wget

Warning

L'-Sopzione che consente di axel utilizzare una http connessione non crittografata per individuare i mirror di un file.

Si consiglia vivamente di migrare qualsiasi utilizzo di axel over verso uno dei due. curl wget

brotlie libbrotli - compressione

I libbrotli pacchetti brotli and erano disponibiliEPEL7, mentre solo il brotli pacchetto era disponibile in AL2 core.

Entrambi i libbrotli pacchetti brotli e sono inclusi in AL2 023.

Il `broTLi` pacchetto può essere installato su AL2 023 con il seguente comando:

```
[ec2-user ~]$ sudo dnf install broTLi
```

Il `libbroTLi` pacchetto può essere installato su AL2 023 con il seguente comando:

```
[ec2-user ~]$ sudo dnf install libbroTLi
```

collectd- Demone per la raccolta delle statistiche

Il `collect` pacchetto era incluso ed era disponibile anche in `collectd` ed `collectd-python3` AL2 Extras. EPEL7

Il `collectd` pacchetto è incluso in AL2 023 e può essere installato eseguendo il seguente comando:

```
[ec2-user ~]$ sudo dnf install collectd
```

cpulimit- Limitatore di utilizzo della CPU

In Amazon Linux 2023, `systemd` fornisce funzionalità per limitare l'utilizzo della CPU di processi o gruppi di processi. Questa funzionalità è facile da usare anche per qualsiasi `systemd` servizio.

Sono disponibili potenti funzionalità di controllo delle risorse `systemd` che possono essere utilizzate per garantire che qualsiasi attività o gruppo di attività abbia un numero limitato di risorse che può consumare. Per ulteriori informazioni, consultate la documentazione upstream di [systemd.resource-control](#), insieme a [Limitazione dell'utilizzo delle risorse di processo in AL2 023 utilizzando systemd](#)

exim- agente di trasferimento della posta

Il `exim` pacchetto era arrivato EPEL7 e precedentemente era disponibile in AL1. Amazon Linux 2023 fornisce sia Mail Transfer Agent () che `sendmail` Mail Transfer Agent (MTAs). `postfix`

fuse3- File system in Userspace (FUSE) v3

Il `fuse3` pacchetto (incluso `fuse3-libs` e `fuse3-devel`) era incluso. EPEL7 Questi pacchetti fanno parte di AL2 023 e ciascuno può essere installato eseguendo il comando seguente pertinente:

```
[ec2-user ~]$ sudo dnf install fuse3
```

```
[ec2-user ~]$ sudo dnf install fuse3-libs
```

```
[ec2-user ~]$ sudo dnf install fuse3-devel
```

ganglia- Sistema di monitoraggio distribuito

Il ganglia pacchetto era in EPEL7 arrivo e precedentemente era disponibile in AL1. Non è stato spedito con AL2.

Il progetto originario ha avuto un periodo di inattività durante il quale alcune lacune non CVEs sono state affrontate. Sebbene vi sia stata un'attività recente nel progetto a monte, non è previsto che si ganglia aggiunga a 023. AL2

git-lfs- controllo delle versioni di file di grandi dimensioni con Git

Il git-lfs pacco era arrivato EPEL7. In Amazon Linux 2023, il git-lfs pacchetto è incluso nel repository principale. Su AL2 023, git-lfs può essere installato eseguendo il seguente comando:

```
[ec2-user ~]$ sudo dnf install git-lfs
```

haveged- una fonte di entropia che utilizza l'algoritmo HAVEGE

Il haveged pacco era arrivato. EPEL7 Amazon Linux 2023 è preconfigurato con fonti di entropia e non richiede l'uso di. haveged

inotify-tools- strumenti da riga di comando di inotify

Il inotify-tools pacchetto era in EPEL7, ed è incluso, in AL2 023.

Note

In AL2 023, systemd supporta l'attivazione basata sul percorso che può essere utilizzata per intervenire su eventi come quando un percorso esiste o cambia.

Gran parte di ciò per cui inotify-tools viene utilizzato ora può essere realizzato meglio in modo più affidabile utilizzando l'attivazione del systemd percorso. Per ulteriori informazioni, vedete [systemd.path](#).

Il `inotify-tools` pacchetto è incluso in AL2 023 e può essere installato eseguendo il comando seguente:

```
[ec2-user ~]$ sudo dnf install inotify-tools
```

iperf- Benchmark TCP/UDP delle prestazioni

Il pacchetto della `iperf` versione 2 era disponibile EPEL7, era disponibile anche in `testing AL2 Extra`. ed era disponibile anche in AL1

Note

Il `iperf3` pacchetto è disponibile anche, che fornisce la versione 3 di `iperf`.

Il `iperf` pacchetto è incluso in AL2 023 e può essere installato eseguendo il comando seguente:

```
[ec2-user ~]$ sudo dnf install iperf
```

jemalloc- implementazione alternativa `malloc`

Il `jemalloc` pacchetto era incluso EPEL7 ed era disponibile in `lamp-mariadb10.2-php7.2` ed `mariadb10.5 AL2 Extras`.

Il `jemalloc` pacchetto è incluso in AL2 023 e può essere installato eseguendo il seguente comando:

```
[ec2-user ~]$ sudo dnf install jemalloc
```

libbsd- Libreria di funzioni compatibile con BSD

Il `libbsd` pacchetto era incluso ed era disponibile anche nell'`Extra`. EPEL7 `testing AL2`

Il `libbsd` pacchetto è incluso in AL2 023 e può essere installato eseguendo il seguente comando:

```
[ec2-user ~]$ sudo dnf install libbsd
```

I file di sviluppo per `libbsd` possono essere installati eseguendo il comando seguente.

```
[ec2-user ~]$ sudo dnf install libbsd-devel
```

libserf- Libreria client HTTP

Il `libserf` pacchetto era arrivato EPEL7. Il `libserf` pacchetto è fornito in Amazon Linux 2023. Può essere installato eseguendo il seguente comando:

```
[ec2-user ~]$ sudo dnf install libserf
```

libzstd- libreria di compressione zstd

Il `libzstd` pacchetto era in AL2 core, oltre che in. EPEL7 Il `libzstd` pacchetto fa anche parte di AL2 023.

```
[ec2-user ~]$ sudo dnf install libzstd
```

lighttpdserver web

Il `lighttpd` pacchetto era in arrivo EPEL7 e in precedenza era disponibile in AL1. Amazon Linux 2023 fornisce sia i server Apache che `httpd` i server `nginx` Web.

lshell- una shell con restrizioni

Il `lshell` pacchetto non è mai stato spedito come parte di Amazon Linux. Era disponibile in EPEL6. Il [repository di pacchetti Fedora lshell](#) spiega [perché non è stato impacchettato](#) in EPEL7 Fedora 30. [È stato anche rimosso da Debian.](#)

[Il lshell progetto originale non viene più mantenuto attivamente e contiene dati critici noti senza patch CVEs: CVE-2016-6902 e CVE-2016-6903.](#)

Anche l'alternativa suggerita nel bug di Debian non è stata mantenuta a monte, e l'autore cita come motivo problemi di sicurezza non risolvibili. [rssh](#)

Per questi motivi, l'aggiunta `lshell` a 023 non è pianificata. AL2

monit- monitoraggio di processi, file, directory e dispositivi

In Amazon Linux 2023, `systemd` offre un'ampia gamma di funzionalità per il monitoraggio, l'avvio, l'arresto e il riavvio dei servizi. Ciò include la limitazione della velocità di riavvio, l'attesa tra i tentativi

di riavvio e l'avvio di un altro servizio in caso di errore. Per ulteriori informazioni, consultate la documentazione di [systemd.service](#).

In AL2 023, supporta systemd anche l'attivazione basata sul percorso, che può essere utilizzata per intervenire su eventi come l'esistenza o le modifiche di un percorso. Per ulteriori informazioni, vedere [systemd.path](#).

Esistono opzioni di configurazione comuni per le systemd unità che consentono di specificare dipendenze, condizionali e azioni da intraprendere in caso di successo o fallimento. [Per ulteriori informazioni, consultate la documentazione di systemd.unit](#).

Sono disponibili potenti funzionalità di controllo delle risorse systemd che possono essere utilizzate per garantire che qualsiasi attività di monitoraggio non utilizzi CPU o memoria eccessive. Per ulteriori informazioni, vedere [systemd.resource-control](#).

nodejs

Il pacchetto della nodejs versione 16 era presente ed nodejs è ora incluso in EPEL7 023. AL2 Al momento della stesura, entrambe le nodejs versioni 18 e 20 erano disponibili nella versione AL2 023. È possibile installare nodejs 18 su AL2 023 con il seguente comando:

```
[ec2-user ~]$ sudo dnf install nodejs
```

È possibile installare nodejs 20 su AL2 023 con il seguente comando:

```
[ec2-user ~]$ sudo dnf install nodejs20
```

perl-Config-General

Il perl-Config-General pacchetto era incluso ed è ora incluso in AL2 023. EPEL7 È possibile installare il perl-Config-General pacchetto in AL2 023 con il seguente comando:

```
[ec2-user ~]$ sudo dnf install perl-Config-General
```

I moduli Perl possono essere installati anche chiedendo di DNF installare il pacchetto che fornisce un particolare modulo Perl. Con questo metodo, è possibile utilizzare il nome del modulo Perl più familiare anziché il nome del pacchetto del sistema operativo.

```
[ec2-user ~]$ sudo dnf install 'perl(Config::General)'
```

python2-lockfile- blocco dei file

Il python2-lockfile pacchetto era arrivato e EPEL7 ne AL2 includeva unopython-lockfile. In AL2 023 [Python 2.7 è stato sostituito con Python 3](#), quindi una variante Python 2 di questo pacchetto non verrà aggiunta AL2 a 023.

La versione Python 3 di questo pacchetto è inclusa in AL2 023. È possibile installare il python3-lockfile pacchetto in AL2 023 con uno dei seguenti comandi:

```
[ec2-user ~]$ sudo dnf install python3-lockfile
```

I moduli Python possono essere installati anche chiedendo di DNF installare il pacchetto che fornisce un particolare modulo Python.

```
[ec2-user ~]$ sudo dnf install 'python3dist(lockfile)'
```

python2-rsa- Python RSA puro

Il python2-rsa pacchetto era AL2 incluso e EPEL7 ne includeva unopython2-rsa. In AL2 023 [Python 2.7 è stato sostituito con Python 3](#), quindi una variante Python 2 di questo pacchetto non verrà aggiunta AL2 a 023.

La versione Python 3 di questo pacchetto è inclusa in AL2 023. È possibile installare il python3-rsa pacchetto in AL2 023 con uno dei seguenti comandi:

```
[ec2-user ~]$ sudo dnf install python3-rsa
```

I moduli Python possono essere installati anche chiedendo di DNF installare il pacchetto che fornisce un particolare modulo Python.

```
[ec2-user ~]$ sudo dnf install 'python3dist(rsa)'
```

python2-simplejson- Routine JSON per Python 2

Il python2-simplejson pacchetto era arrivato. EPEL7 In AL2 023 [Python 2.7 è stato sostituito con Python 3](#), quindi una variante Python 2 di questo pacchetto non verrà aggiunta AL2 a 023.

La versione Python 3 di questo pacchetto è inclusa in AL2 023. È possibile installare il python3-simplejson pacchetto in AL2 023 con il seguente comando:

```
[ec2-user ~]$ sudo dnf install python3-simplejson
```

I moduli Python possono essere installati anche chiedendo di DNF installare il pacchetto che fornisce un particolare modulo Python.

```
[ec2-user ~]$ sudo dnf install 'python3dist(simplejson)'
```

rkhunter- Rootkit Hunter

Il `rkhunter` pacchetto è incluso in AL2 023 insieme a `chkrootkit`

```
[ec2-user ~]$ sudo dnf install rkhunter
```

```
[ec2-user ~]$ sudo dnf install chkrootkit
```

rssh- una shell limitata da usare con OpenSSH

Il `rssh` pacchetto era arrivato. EPEL7 Il [rssh](#) pacchetto originale non è sottoposto a manutenzione e l'autore cita come motivo problemi di sicurezza non risolvibili.

Poiché l'autore cita problemi di sicurezza irrisolvibili, non è prevista l'aggiunta a 023. `rssh` AL2

sscg- generatore di certificati SSL autofirmati

Il `sscg` pacchetto era integrato AL2 , oltre che integrato. EPEL7 Il `sscg` pacchetto fa anche parte di AL2 023.

```
[ec2-user ~]$ sudo dnf install sscg
```

stress- Test di stress

Il `stress` pacchetto era arrivato EPEL7, ed era disponibile anche in AL1

Il `stress` pacchetto è incluso in AL2 023 e può essere installato eseguendo il seguente comando:

```
[ec2-user ~]$ sudo dnf install stress
```

stress-ng- Test di stress

Il `stress-ng` pacchetto era arrivato EPEL7 ed era disponibile anche nella versione `testing` AL2 Extra.

Il `stress-ng` pacchetto è incluso in AL2 023 e può essere installato eseguendo il seguente comando:

```
[ec2-user ~]$ sudo dnf install stress-ng
```

tmpwatch- rimuove i file in base all'ora dell'ultimo accesso

In Amazon Linux 2023, questa funzionalità è fornita da [systemd-tmpfiles](#).

xmlstarlet- utilità XML da riga di comando

Il `xmlstarlet` pacchetto era incluso e non è disponibile nella versione AL2 023. EPEL7

Il pacchetto upstream non viene modificato da oltre 9 anni (l'ultima modifica risale ad agosto 2014). Per altri quattro anni precedenti (almeno da luglio 2010), una richiesta per un nuovo manutentore è rimasta senza risposta. È per questo motivo che non è prevista l'aggiunta `xmlstarlet` a 023. AL2

Python 2.7 è stato sostituito con Python 3

AL2 fornisce supporto e patch di sicurezza per Python 2.7 fino a giugno 2025, come parte del nostro impegno di supporto a lungo termine (LTS) per i pacchetti principali. AL2 Questo supporto si estende oltre la dichiarazione della comunità Python upstream di Python end-of-life 2.7 di gennaio 2020.

AL2 usa il gestore di yum pacchetti, che ha una forte dipendenza da Python 2.7. Nella AL2 versione 023 il gestore di dnf pacchetti è migrato a Python 3 e non richiede più Python 2.7. AL2023 è completamente passato a Python 3.

Note

AL2023 ha rimosso Python 2.7, quindi tutti i componenti del sistema operativo che richiedono Python sono scritti per funzionare con Python 3. Per continuare a usare una versione di Python fornita e supportata da Amazon Linux, converti il codice di Python 2 in Python 3.

Per ulteriori informazioni su Python su Amazon Linux, consulta [Pythonnel AL2023](#).

Aggiornamenti di sicurezza

Amazon Linux 2023 migliora l'indurimento presente in AL2. Per ulteriori informazioni, consulta [Sicurezza e conformità in Amazon Linux 2023](#). Per ulteriori informazioni sulle modifiche all'hardening del kernel da AL2, consulta. [Modifiche alla configurazione del kernel incentrate sulla sicurezza](#)

Argomenti

- [SELinux](#)
- [OpenSSL 3](#)
- [IMDSv2](#)
- [Rimozione di hotpatch log4j \(log4j-cve-2021-44228-hotpatch\)](#)

SELinux

Per impostazione predefinita, Security Enhanced Linux (SELinux) per AL2 023 è impostato su enabled mode. `permissive` In modalità `permissive`, le negazioni di autorizzazione vengono registrate ma non applicate.

SELinux è una funzionalità di sicurezza del kernel Amazon Linux, che era `disabled` in AL2. SELinux è una raccolta di funzionalità e utilità del kernel che fornisce l'architettura obbligatoria di controllo degli accessi (MAC) nei principali sottosistemi del kernel.

Per ulteriori informazioni, consulta [Impostazione SELinux delle modalità per AL2 023](#).

[Per ulteriori informazioni su SELinux repository, strumenti e policy, consulta SELinux Notebook, Types of SELinux policy e Project. SELinux](#)

OpenSSL 3

AL2023 include il toolkit di Open Secure Sockets Layer version 3 (OpenSSL 3) crittografia. AL2023 supporta TLS 1.3 e protocolli di rete. TLS 1.2

Per impostazione predefinita, AL2 viene fornito con OpenSSL 1.0.2. Puoi sviluppare applicazioni contro OpenSSL 1.1.1.

Per ulteriori informazioni su OpenSSL, consulta la [guida per la migrazione a OpenSSL](#).

Per ulteriori informazioni sulla sicurezza, consulta [Aggiornamenti e funzionalità di sicurezza](#).

IMDSv2

Per impostazione predefinita, tutte le istanze avviate con l' AL2AMI 023 richiedono IMDSv2 - only e il limite di hop predefinito sarà impostato su 2 per consentire il supporto di carichi di lavoro containerizzati. Questo è possibile impostando il parametro `imds-support` su `v2.0`. Per ulteriori informazioni, consulta [Configurare l'AMI](#) nella Amazon EC2 User Guide.

Note

Il periodo di validità del token di sessione può essere compreso tra 1 secondo e 6 ore. Gli indirizzi a cui inviare le richieste API per le query IMDSv2 sono i seguenti:

- IPv4: 169254,169254
- IPv6: fd00:ec2: :254

È possibile sovrascrivere manualmente queste impostazioni e abilitarle IMDSv1 utilizzando le proprietà di avvio dell'opzione Instance Metadata. Puoi anche utilizzare i controlli IAM per applicare impostazioni diverse. IMDS Per ulteriori informazioni sulla configurazione e l'utilizzo del servizio di metadati delle istanze, consulta [Usa IMDSv2, configura le opzioni dei metadati delle istanze per le nuove istanze e Modifica delle opzioni dei metadati delle istanze per le istanze esistenti nella Amazon User Guide](#). EC2

Rimozione di hotpatch log4j (**log4j-cve-2021-44228-hotpatch**)

Note

AL2023 non viene spedito con il pacco. `log4j-cve-2021-44228-hotpatch`

[In risposta a CVE-2021-44228, Amazon Linux ha rilasciato una versione in pacchetto RPM di Hotpatch per Apache Log4j per e. AL1 AL2](#) Nell'[annuncio dell'aggiunta dell'hotpatch ad Amazon Linux](#) abbiamo indicato che l'installazione dell'hotpatch non sostituisce l'aggiornamento a una versione log4j che mitiga CVE-2021-44228 o CVE-2021-45046.

L'hotpatch era una mitigazione per consentire il tempo necessario per applicare le patch `log4j`. La prima versione General Availability (GA) di 023 risale a 15 mesi dopo [CVE-2021-44228](#), AL2 quindi 023 non viene fornito con l'hotpatch (abilitato o meno). AL2

[Gli utenti che eseguono log4j le proprie versioni su Amazon Linux devono assicurarsi di aver effettuato l'aggiornamento alle versioni non interessate da CVE-2021-44228 o CVE-2021-45046.](#)

AL2023 fornisce indicazioni su come tenerti aggiornato sulle patch di sicurezza. [Aggiornamento AL2023](#) Gli avvisi di sicurezza sono pubblicati nella pagina [Amazon Linux Security Center](#).

Aggiornamenti deterministici per la stabilità

Con gli aggiornamenti deterministici tramite la funzionalità di repository con versioni, ogni AL2 AMI 023 per impostazione predefinita è bloccata su una versione specifica del repository. Puoi usare gli aggiornamenti deterministici per ottenere una maggiore coerenza tra gli aggiornamenti e le versioni dei pacchetti. Ogni rilascio, principale o secondario, include una versione di repository specifica.

Novità della versione AL2 023, l'aggiornamento deterministico è abilitato per impostazione predefinita. Si tratta di un miglioramento rispetto al metodo di blocco manuale e incrementale utilizzato in altre versioni precedenti. AL2

Per ulteriori informazioni, consulta [Aggiornamenti deterministici tramite repository con versioni su AL2023](#).

gp3 come tipo di volume Amazon EBS predefinito

L'AMI AL2 023 ed AL2 entrambe utilizzano il XFS file system sul file system root. Per AL2 023, le `mkfs` opzioni per il file system del dispositivo root sono ulteriormente ottimizzate per Amazon EC2. AL2023 supporta anche una serie di altri file system che puoi utilizzare su altri volumi per soddisfare i tuoi requisiti specifici.

AL2023 AMIs utilizza i gp3 volumi Amazon EBS per impostazione predefinita, mentre utilizza i gp2 volumi AL2 AMIs Amazon EBS per impostazione predefinita. Puoi modificare il tipo di volume quando avvii un'istanza.

Per ulteriori informazioni sui tipi di volume di Amazon EBS, consulta la pagina dedicata ai [volumi per uso generale di Amazon EBS](#).

Per ulteriori informazioni sul lancio di un' EC2 istanza Amazon, consulta [Launch an instance](#) nella Amazon EC2 User Guide.

Gerarchia dei gruppi di controllo unificati (cgroup v2)

Un gruppo di controllo (cgroup) è una funzionalità del kernel Linux per organizzare gerarchicamente i processi e distribuire le risorse di sistema tra di essi. I gruppi di controllo vengono usati estensivamente per implementare un runtime di container e da `systemd`.

AL2 supporta `cgroupv1` e AL2 023 supporta `cgroupv2`. Ciò è particolarmente importante se si eseguono carichi di lavoro containerizzati, ad esempio in caso di [Utilizzo di Amazon ECS basato su AL2 023 AMIs per ospitare carichi di lavoro containerizzati](#).

Sebbene AL2 023 includa ancora codice che può far funzionare il sistema utilizzando `cgroupv1`, questa non è una configurazione consigliata o supportata e verrà completamente rimossa in una futura versione principale di Amazon Linux.

Esiste un'ampia documentazione riguardante le [interfacce del kernel Linux di basso livello](#), nonché la [documentazione sulla delega `systemd` cgroup](#).

Un caso d'uso comune al di fuori dei contenitori consiste nella creazione di `systemd` unità con limiti alle risorse di sistema che possono utilizzare. Per ulteriori informazioni, vedere [systemd.resource-control](#).

`systemd` timer sostituiscono `cron`

Il `cronie` pacchetto è stato installato di default sull' AL2 AMI, fornendo supporto per il `crontab` metodo tradizionale di pianificazione delle attività periodiche. In AL2 023, non `cronie` è incluso per impostazione predefinita. Pertanto, il supporto per non `crontab` è più fornito per impostazione predefinita.

Facoltativamente, è possibile installare il pacchetto `cronie` per usare i processi `cron` classici. Ti consigliamo di eseguire la migrazione ai timer `systemd` grazie alle funzionalità aggiuntive fornite da `systemd`.

Toolchain migliorata: `gccbinutils`, e `glibc`

AL2023 include molti degli stessi pacchetti principali di AL2

Abbiamo aggiornato i seguenti tre pacchetti di toolchain principali per AL2 023.

Nome pacchetto	AL2	AL2023
glibc	2,26	2,34
gcc	7.3	11,3
binutils	2,29	2,39

Per ulteriori informazioni, consulta [Pacchetti di toolchain principali glibc, gcc, binutils](#).

Per ulteriori informazioni sui tempi di esecuzione Fortran dei linguaggi CC++, inclusi gli standard linguistici predefiniti aggiornati, vedere. [CC++, e Fortran nel AL2 2023](#)

Per ulteriori informazioni su queste ottimizzazioni, consulta [Ottimizzazioni relative a operazioni e prestazioni](#).

systemd il diario sostituisce rsyslog

Nel AL2 023 il pacchetto del sistema di registrazione è cambiato da. AL2 AL2023 non viene installato rsyslog per impostazione predefinita, quindi i file di registro basati su testo come /var/log/messages quelli disponibili in AL2 non sono disponibili per impostazione predefinita. La configurazione predefinita per AL2 023 è systemd-journal, che può essere esaminata utilizzando journalctl. Sebbene rsyslog sia un pacchetto opzionale in AL2 023, consigliamo la nuova journalctl interfaccia systemd basata e i pacchetti correlati. Per ulteriori informazioni, consulta la pagina del manuale di [journalctl](#).

L'equivalente di alcuni syslog comandi di uso comune è riportato nella tabella seguente.

Comando AL2 syslog	AL2equivalente a 023 systemd journal
<pre>[ec2-user ~]\$ cat /var/log/messages</pre>	<pre>[ec2-user ~]\$ journalctl</pre>
<pre>[ec2-user ~]\$ tail -f /var/log/messages</pre>	<pre>[ec2-user ~]\$ journalctl -f</pre>

Comando AL2 syslog	AL2equivalente a 023 systemd journal
<pre>[ec2-user ~]\$ grep foo /var/log/messages</pre>	<pre>[ec2-user ~]\$ journalctl grep foo</pre>

Dipendenze dei pacchetti ridotte al minimo

Amazon Linux 2023 riduce al minimo il grafico delle dipendenze di molti pacchetti per fornire un ingombro ridotto per le applicazioni. Le principali modifiche apportate AL2 includono i `gnupg-minimal` pacchetti `curl-minimal` and, che riducono in modo significativo il numero di pacchetti richiesti pur mantenendo le funzionalità di uso comune.

Argomenti

- [Modifiche ai pacchetti per curl e libcurl](#)
- [GNU Privacy Guard \(GNUPG\)](#)

Modifiche ai pacchetti per **curl** e **libcurl**

AL2023 separa i protocolli e le funzionalità comuni dei pacchetti e li divide in `e. curl libcurl curl-minimal libcurl-minimal` Ciò riduce l'ingombro su disco, memoria e dipendenze per la maggior parte degli utenti ed è il pacchetto predefinito per AL2 023 e contenitori. AMLs

Se è richiesta la piena funzionalità di `curl`, ad esempio per il supporto di `gopher://`, esegui i seguenti comandi per installare i pacchetti `curl-full` e `libcurl-full`.

```
$ dnf swap libcurl-minimal libcurl-full
```

```
$ dnf swap curl-minimal curl-full
```

GNU Privacy Guard (GNUPG)

AL2023 separa le funzionalità minime da quelle complete per il `gnupg2` pacchetto e i pacchetti. `gnupg2-minimal` `gnupg2-full` Per impostazione predefinita, solo il pacchetto `gnupg2-minimal` viene installato. In questo modo è possibile ottenere la funzionalità minima richiesta per verificare le firme digitali sui pacchetti `rpm`.

Per ulteriori funzionalità di gnupg2 (ad esempio la possibilità di scaricare le chiavi da un server di chiavi), assicurati che il pacchetto `gnupg2-full` sia installato. Esegui il comando riportato di seguito per scambiare `gnupg2-minimal` e `gnupg2-full`.

```
$ dnf swap gnupg2-minimal gnupg2-full
```

Amazon Corretto come JVM predefinita

AL2023 viene fornito con [Amazon Corretto](#) come Java Development Kit (JDK) predefinito (e unico). Tutti i pacchetti Java basati in AL2 023 sono tutti creati con Amazon Corretto 17.

Se stai migrando da AL2, puoi passare senza problemi dalla OpenJDK versione equivalente a AL2 Amazon Corretto.

AWS CLIv2

AL2023 viene fornito con AWS CLI la versione 2, mentre AL2 viene fornito con la versione 1 di AWS CLI.

Avvio UEFI preferito e sicuro

Per impostazione predefinita, tutte le istanze avviate con l'AL2AMI 023 su tipi di istanza che supportano il firmware UEFI verranno avviate in modalità UEFI. Questo è possibile impostando il parametro della modalità di avvio dell'AMI su `uefi-preferred`. Per ulteriori informazioni, consulta le [modalità di avvio](#) nella Amazon EC2 User Guide.

Sui tipi di EC2 istanze Amazon che supportano UEFI Secure Boot, è possibile abilitare Secure Boot in Amazon Linux 2023. Per ulteriori informazioni, consulta [Avvio sicuro UEFI attivo AL2023](#).

Modifiche alla configurazione predefinita del server SSH

Per l'AMI AL2 023, abbiamo cambiato i tipi di chiavi `sshd` host che generiamo con la versione. Abbiamo anche eliminato alcuni tipi di chiavi legacy per evitare di generarli al momento dell'avvio. I client devono supportare i protocolli `rsa-sha2-256` e `rsa-sha2-512` oppure `ssh-ed25519` con l'uso di una chiave `ed25519`. Per impostazione predefinita, le firme `ssh-rsa` sono disabilitate.

Inoltre, le impostazioni di configurazione AL2 023 nel `sshd_config` file predefinito contengono `UseDNS=no`. Questa nuova impostazione significa che è meno probabile che eventuali problemi con

DNS blocchiano la capacità di stabilire sessioni ssh con le istanze. Come compromesso, le voci di riga `from=hostname.domain,hostname.domain` nei file `authorized_keys` non verranno risolte. Poiché `sshd` non tenta più di risolvere i nomi DNS, ogni valore `hostname.domain` separato da virgole deve essere tradotto in un IP address corrispondente.

Per ulteriori informazioni, consulta [Configurazione del server SSH predefinita](#).

AL2023 modifiche al kernel da AL2

AL2023 include il kernel 6.1 e molte modifiche alla configurazione per ottimizzare ulteriormente Amazon Linux per il cloud. Per la maggior parte degli utenti, queste modifiche dovrebbero essere completamente trasparenti.

IPv4 TTL

Il TTL per IPv4 è configurato tramite `sysctl`, con i valori predefiniti presenti in `/etc/sysctl.d/00-defaults.conf`. Questo valore può essere personalizzato con i `sysctl` metodi usuali. Per ulteriori informazioni, consulta la `sysctl` man pagina.

AL2 imposta il `net.ipv4.ip_default_ttl` valore su 255, mentre lo AL2023 imposta su 127. Ciò allinea le impostazioni predefinite di Amazon Linux a quelle delle altre principali distribuzioni Linux. Non è consigliabile modificare questa impostazione predefinita senza una dimostrata necessità.

Modifiche alla configurazione del kernel incentrate sulla sicurezza

Opzione	AL2/4.1	AL2/4.1	AL2/5.1	AL2/5.1	AL2023	AL2023	AL2023	AL2023	AL2023	AL2023/6.
CONFIG	aarch64	x86_64	aarch64	x86_64	1/ aarch64	1/ x86_64	12/ aarch64	12/ x86_64	18/ aarch64	18/ x86_64
CONFIG	n	y	n	y	y	y	y	y	y	y
G_ON_D										
_CORRU										
ON										
CONFIG	4096	4096	4096	4096	65536	65536	65536	65536	65536	65536
FAULT_										

Opzione	AL2/4.1	AL2/4.1	AL2/5.1	AL2/5.1	AL2023	AL2023	AL2023	AL2023	AL2023	AL2023/6.
CONFIG	aarch64	x86_64	aarch64	x86_64	1/ aarch64	1/ x86_64	12/ aarch64	12/ x86_64	18/ aarch64	18/ x86_64
<u>P_MIN_R</u>										
<u>CONFIG_VMEM</u>	n	y	n	y	n	n	n	n	n	n
<u>CONFIG_VPORT</u>	n	y	n	y	n	n	n	n	n	n
<u>CONFIG_RTIFY_RCE</u>	n	y	n	y	y	y	y	y	y	y
<u>CONFIG_RDENED_ERCOPY_LLBACK</u>	N/D	N/D	y	y	N/D	N/D	N/D	N/D	N/D	N/D
<u>CONFIG_IT_ON_OC_DEFINITION_ON</u>	N/D	N/D	n	n	n	n	n	n	n	n
<u>CONFIG_IT_ON_E_DEFAULT_ON</u>	N/D	N/D	n	n	n	n	n	n	n	n
<u>CONFIG_MMU_DEFLT_DMA_RICT</u>	N/D	N/D	N/D	N/D	n	n	n	n	n	n

Opzione	AL2/4.1	AL2/4.1	AL2/5.1	AL2/5.1	AL2023	AL2023	AL2023	AL2023	AL2023	AL2023/6.
CONFIG	aarch64	x86_64	aarch64	x86_64	1/ aarch64	1/ x86_64	12/ aarch64	12/ x86_64	18/ aarch64	18/ x86_64
CONFIG ISC_AL OAD	y	y	y	y	n	n	n	n	n	n
CONFIG HED_CC	N/D	N/D	N/D	N/D	N/D	y	N/D	y	N/D	y
CONFIG HED_ST _END_C K	n	y	n	y	y	y	y	y	y	y
CONFIG CURITY ESG_RE ICT	n	n	n	n	y	y	y	y	y	y
CONFIG CURITY LINUX_ ABLE	y	y	y	y	n	n	N/D	N/D	N/D	N/D
CONFIG UFFLE_ E_ALLC OR	N/D	N/D	y	y	y	y	y	y	y	y
CONFIG AB_FRE ST_HAR ED	n	y	y	y	y	y	y	y	y	y

Opzione CONFIG	AL2/4.1 aarch64	AL2/4.1 x86_64	AL2/5.1 aarch64	AL2/5.1 x86_64	AL2023 1/ aarch64	AL2023 1/ x86_64	AL2023 12/ aarch6 4	AL2023 12/ x86_64	AL2023 18/ aarch6 4	AL2023/6. 18/ x86_64
CONFIG_AB_FRE ST_RAN	n	n	y	y	y	y	y	y	y	y

Modifiche alla configurazione del kernel incentrate sulla sicurezza specifiche per x86-64

Opzione CONFIG	AL2/4.14/ x86_64	AL2/5.10/ x86_64	AL2023/6.1/ x86_64	AL2023/6.12/ x86_64	AL2023/6.18/ x86_64
CONFIG_AM D_IOMMU	y	y	y	y	y
CONFIG_AM D_IOMMU_V 2	m	m	y	N/D	N/D
CONFIG_RA NDOMIZE_M EMORY	N/D	y	y	y	y

Modifiche alla configurazione del kernel incentrate sulla sicurezza specifiche per aarch64 (ARM/Graviton)

Opzione CONFIG	AL2/4.14/ aarch64	AL2/5.10/ aarch64	AL2023/6.1/ aarch64	AL2023/6.12/ aarch64	AL2023/6.18/ aarch64
CONFIG_ARM64_PTR_AUTH	N/D	y	y	y	y
CONFIG_ARM64_PTR_AUTH_KERNEL	N/D	N/D	y	y	y
CONFIG_ARM64_SW_TTBR0_PAN	y	y	y	y	y

`/dev/mem`, `/dev/kmem` e `/dev/port`

Amazon Linux 2023 disabilita `/dev/mem` e `/dev/port` (`CONFIG_DEVMEM`/`CONFIG_DEVPORT`) completamente, si basa sulle restrizioni già in vigore in AL2.

Il `/dev/kmem` codice è stato completamente rimosso da Linux nel kernel 5.13 e, sebbene fosse disabilitato in AL2, ora non è applicabile a AL2023.

Questa opzione è una delle [impostazioni consigliate per Kernel Self Protection Project](#).

FORTIFY_SOURCE

AL2023 abilita `CONFIG_FORTIFY_SOURCE` su tutte le architetture supportate. Questa funzionalità mira al rafforzamento della sicurezza. Laddove il compilatore è in grado di determinare e convalidare le dimensioni del buffer, questa funzionalità consente di rilevare gli overflow del buffer nelle funzioni comuni di stringa e memoria.

Questa opzione è una delle [impostazioni consigliate per Kernel Self Protection Project](#).

Autoload di Line Discipline () **CONFIG_LDISC_AUTOLOAD**

Il AL2023 kernel non caricherà automaticamente le discipline di riga, ad esempio mediante software che utilizza il `TIOCSETDioct1`, a meno che la richiesta non provenga da un processo con i permessi necessari. `CAP_SYS_MODULE`

Questa opzione è una delle [impostazioni consigliate per Kernel Self Protection Project](#).

dmesgaccesso per utenti non privilegiati () **CONFIG_SECURITY_DMESG_RESTRICT**

Per impostazione predefinita, AL2023 non consente agli utenti non privilegiati di accedere a. `dmesg`

Questa opzione è una delle [impostazioni consigliate per Kernel Self Protection Project](#).

SELinux **selinuxfs**disabilitare

AL2023 disattiva l'opzione obsoleta del `CONFIG_SECURITY_SELINUX_DISABLE` kernel, che abilitava un metodo di disabilitazione in fase di runtime prima del caricamento della policy SELinux .

Questa opzione è una delle [impostazioni consigliate per Kernel Self Protection Project](#).

Altre modifiche apportate alla configurazione del kernel

Opzione	AL2/4.1	AL2/4.1	AL2/5.1	AL2/5.1	AL2023	AL2023	AL2023	AL2023	AL2023	AL2023/6.
CONFIG	aarch64	x86_64	aarch64	x86_64	1/ aarch64	1/ x86_64	12/ aarch6 4	12/ x86_64	18/ aarch6 4	18/ x86_64
<u>CONFIG</u>	100	250	100	250	100	100	100	100	100	100
<u>CONFIG</u>	4096	8192	4096	8192	4096	8192	4096	8192	4096	8192
<u>CONFIG</u>	y	n	y	n	y	y	y	y	y	y
<u>CONFIG</u>										
<u>CONFIG</u>										
<u>CONFIG</u>										
<u>CONFIG</u>										

Opzione	AL2/4.1	AL2/4.1	AL2/5.1	AL2/5.1	AL2023	AL2023	AL2023	AL2023	AL2023	AL2023/6.
CONFIG	aarch64	x86_64	aarch64	x86_64	1/ aarch64	1/ x86_64	12/ aarch64	12/ x86_64	18/ aarch64	18/ x86_64
CONFIG_NIC_ONPS_VAL	1	0	1	0	1	1	1	1	N/D	N/D
CONFIG_P	m	m	m	m	m	m	m	m	m	m
CONFIG_IP	m	m	m	m	n	n	n	n	n	n
CONFIG_N_PV	N/D	y	N/D	n	N/D	n	N/D	n	N/D	n

CONFIG_HZ

AL2023 impostato CONFIG_HZ su 100 su entrambe le piattaforme. x86-64 aarch64

CONFIG_NR_CPUS

AL2023 imposta CONFIG_NR_CPUS un numero più vicino al numero massimo di core CPU presenti in Amazon EC2.

Panic su OOPS

Il AL2023 kernel andrà in panico quando si interrompe. Questa funzionalità equivale all'avvio con `oops=panic` dalla riga di comando del kernel.

Un oops del kernel si verifica quando il kernel rileva un errore interno che può influire sull'ulteriore affidabilità del sistema.

Supporto per PPP e SLIP

AL2023 non supporta il protocollo SLIP ma può supportare il protocollo PPP nei kernel più recenti. AL2023

Supporto per guest Xen PV

AL2023 non supporta l'esecuzione come guest Xen PV.

Supporto per i file system del kernel

Sono state apportate diverse modifiche ai file system che il kernel AL2 supporterà il montaggio, oltre a cambiamenti negli schemi di partizionamento che il kernel analizzerà.

Opzione CONFIG	AL2/4.1 aarch64	AL2/4.1 x86_64	AL2/5.1 aarch64	AL2/5.1 x86_64	AL2023 1/ aarch64	AL2023 1/ x86_64	AL2023 12/ aarch64	AL2023 12/ x86_64	AL2023 18/ aarch64	AL2023/6. 18/ x86_64
CONFIG_S_FS	n	m	n	m	n	n	n	n	n	n
CONFIG_RRPC	n	m	n	m	n	n	n	n	n	n
CONFIG_D_DISK_EL	y	y	y	y	n	n	n	n	n	n
CONFIG_AMFS	m	m	m	m	n	n	n	n	n	n
CONFIG_AMFS_BKDEV	N/D	N/D	y	n	N/D	N/D	N/D	N/D	N/D	N/D
CONFIG_CLONE	N/D	N/D	n	n	n	n	n	n	n	n
CONFIG_ERA	m	n	m	n	n	n	n	n	n	n

Opzione CONFIG	AL2/4.1 aarch64	AL2/4.1 x86_64	AL2/5.1 aarch64	AL2/5.1 x86_64	AL2023 1/ aarch64	AL2023 1/ x86_64	AL2023 12/ aarch6 4	AL2023 12/ x86_64	AL2023 18/ aarch6 4	AL2023/6. 18/ x86_64
<u>CONFIG_INTEGRITY</u>	n	m	n	m	m	m	m	m	m	m
<u>CONFIG_LOG_WATCHES</u>	n	n	m	m	m	m	m	m	m	m
<u>CONFIG_SWIOTLB</u>	m	n	m	n	n	n	n	n	n	n
<u>CONFIG_VERIFY</u>	m	n	m	n	m	m	m	m	m	m
<u>CONFIG_ZIPFS</u>	n	m	n	m	n	n	n	n	n	n
<u>CONFIG_FAT_FS</u>	N/D	N/D	m	m	m	m	m	m	m	m
<u>CONFIG_T2_FS</u>	n	m	n	m	n	n	n	n	n	n
<u>CONFIG_T3_FS</u>	n	m	n	m	n	n	n	n	N/D	N/D
<u>CONFIG_S2_FS</u>	m	m	m	m	n	n	n	n	n	n
<u>CONFIG_SPLUS_</u>	n	m	n	m	n	n	n	n	n	n
<u>CONFIG_S_FS</u>	n	m	n	m	n	n	n	n	n	n

Opzione CONFIG	AL2/4.1 aarch64	AL2/4.1 x86_64	AL2/5.1 aarch64	AL2/5.1 x86_64	AL2023 1/ aarch64	AL2023 1/ x86_64	AL2023 12/ aarch64	AL2023 12/ x86_64	AL2023 18/ aarch64	AL2023/6. 18/ x86_64
<u>CONFIG_S_FS</u>	n	n	n	n	n	n	n	n	n	n
<u>CONFIG_M_PARTITION</u>	n	y	n	y	n	n	n	n	n	n
<u>CONFIG_C_PARTITION</u>	n	y	n	y	n	n	n	n	n	n
<u>CONFIG_S_V2</u>	n	m	n	m	n	n	n	n	n	n
<u>CONFIG_FS_FS</u>	n	m	n	n	n	n	n	n	n	n
<u>CONFIG_MFS_FS</u>	n	m	n	m	n	n	n	n	n	n
<u>CONFIG_LARIS_PARTITION</u>	n	y	n	y	n	n	n	n	n	n
<u>CONFIG_UASHFSTD</u>	n	y	n	y	y	y	y	y	y	y
<u>CONFIG_N_PARTITION</u>	n	y	n	y	n	n	n	n	n	n

Supporto per Andrew File System (AFS)

Il kernel non è più compilato con il supporto per il file system `afs`. `afs` AL2 non è stato fornito con il supporto dello spazio utente per `afs`.

Supporto per `cramfs`

Il kernel non è più compilato con il supporto per il file system `cramfs`. Il successore di AL2023 è il `squashfs` file system.

Supporto per BSD disklabel

Il kernel non è più compilato con il supporto per le etichette del disco BSD. Se è necessaria la lettura di volumi con etichette su disco BSD, è possibile avviare diversi BSDs.

Modifiche a Device Mapper

Sono state apportate diverse modifiche ai target Device Mapper configurati nel kernel AL2023.

eCryptFs supporto

Il file system `ecryptfs` è stato dichiarato obsoleto in Amazon Linux. I componenti dello spazio utente di `ecryptfs` erano presenti in AL1, rimossi e AL2023 non compila più il kernel con supporto. `AL2` `ecryptfs`

exFAT

Il supporto per il `exFAT` file system è stato aggiunto nel kernel 5.10. AL2 Non era presente al AL2 lancio con un kernel 4.14. AL2023 continua a supportare il `exFAT` file system.

File system `ext2`, `ext3` e `ext4`

AL2023 viene fornito con l'opzione `CONFIG_EXT4_USE_FOR_EXT2`, il che significa che il codice del `ext4` file system verrà utilizzato per leggere i `ext2` file system precedenti.

`CONFIG__FS_GFS2`

Il kernel non è più compilato con `CONFIG__FS_GFS2`.

Supporto per il file system Apple HFS Extended (HFS+)

Nel AL2, solo i x86-64 kernel sono stati compilati con il supporto del `hfsplus` file system. Il kernel AL2 5.15 non include il `hfsplus` supporto su nessuna architettura. Nel AL2023, completiamo l'obsolescenza del supporto `hfsplus` in Amazon Linux.

Supporto per il file system HFS

Nel AL2, solo i x86-64 kernel sono stati creati con il supporto del `hfs` file system. Il kernel AL2 5.15 non include il `hfs` supporto su nessuna architettura. Nel AL2023, completiamo l'obsolescenza del supporto `hfs` in Amazon Linux.

Supporto per il file system JFS

I AL2 x86-64 kernel più vecchi venivano creati con `jfs` il supporto per i file system. Il kernel AL2 5.15 non include il `jfs` supporto su nessuna architettura. Nessuno dei due AL1 o AL2 fornito con JFS userspace. Nel AL2023, completiamo l'obsolescenza del supporto `jfs` in Amazon Linux.

[Il kernel Linux upstream sta valutando la rimozione di JFS](#) Pertanto, se avete dati su un JFS file system, dovrete migrarli su un altro file system. Nel 2024, JFS è stato rimosso da tutti gli attuali kernel Amazon Linux.

WindowsSupporto per Logical Disk Manager (Dynamic Disk) ()

CONFIG_LDM_PARTITION

AL2023 non supporta più Windows 2000 i Windows XP dischi Windows Vista dinamici con partizioni di MS-DOS stile. Questo codice non ha mai supportato i nuovi dischi dinamici basati su GPT introdotti con Windows Vista

Supporto per la mappa di partizione Macintosh

AL2023 non supporta più la classica mappa delle partizioni Macintosh. Le versioni recenti di macOS creeranno per impostazione predefinita tabelle di partizione GPT più moderne rispetto a questo tipo precedente.

NFSv2 supporto

AL2023 non supporta più NFSv2, ma continua a supportare NFSv3 NFSv4, NFSv4 .1 e NFSv4 .2. Ti consigliamo di eseguire la migrazione a NFSv3 o versioni successive.

NTFS (**CONFIG_NTFS_FS**)

Il `ntfs3` codice è stato sostituito `ntfs` per l'accesso ai file system NTFS su Amazon Linux a partire dal kernel 5.10 in. AL2 AL2023 non include più il `ntfs` codice e si basa esclusivamente sul `ntfs3` codice per accedere ai file system NTFS.

File system `romfs`

Il `squashfs` file system è il successore del `romfs` file system di Amazon Linux e il AL2023 kernel non è più costruito con il supporto per `romfs`.

Formato di partizione del disco rigido Solaris x86

AL2023 non supporta più il formato di partizione del disco rigido Solaris x86.

squashfszstdcompressione

AL2023 aggiunge il supporto per i `squashfs` file system `zstd` compressi su tutte le architetture supportate.

Supporto per tabella di partizione Sun

AL2023 non include più il supporto per il formato della tabella delle partizioni Sun (`CONFIG_SUN_PARTITION`).

/tmp è ora `tmpfs`

Amazon Linux 2023 introduce modifiche al `/tmp` comportamento rispetto ad Amazon Linux 2. La configurazione predefinita AL2 era che entrambi `/tmp` e `/var/tmp` trovavano nel file system root. Amazon Linux 2023 utilizza `tmpfs` per `/tmp` impostazione predefinita un limite del 50% di RAM e un massimo di un milione inodes. Queste modifiche allineano Amazon Linux al comportamento di altre distribuzioni Linux.

Per i dettagli completi sul layout del file system di AL2 023, consulta [/tmp](#) e [/var/tmp](#) nella [Layout del file system](#) sezione.

Modifiche all'AMI e all'immagine del contenitore

Sono state apportate alcune modifiche ai pacchetti inclusi AMIs e ai contenitori.

Amazon Linux 2023 introduce un e [the section called “AL2023 Immagine minima del contenitore”](#) il supporto per la creazione [the section called “Creazione di immagini di contenitori essenziali AL2023”](#). Per ulteriori informazioni, consulta [Utilizzo AL2023 in contenitori](#).

Confronto dei pacchetti installati su Amazon Linux 2 e Amazon Linux 2023 AMIs

Un confronto tra il RPMs presente sullo standard AMIs Amazon Linux 2 e AL2 023.

Pacchetto	AL2 AMI	AL2023 AMI
acl	2,2,51	2.3.1
acpid	2.0,19	2.0,32
alternatives		1.15
amazon-chrony-config		4.3
amazon-ec2-net-utils		2.5.1
amazon-linux-extras	2.0.3	
amazon-linux-extras-yum-plugin	2.0.3	
amazon-linux-repo-s3		2023,620241031
amazon-linux-sb-keys		2023,1
amazon-rpm-config		228
amazon-ssm-agent	3,3,987,0	3,3,987,0
amd-ucode-firmware	20200421 (marzo)	20210208 (nomarzo)
at	3.1.13	3,1,23
attr	2,4,46	2.5.1

Pacchetto	AL2 AMI	AL2023 AMI
audit	28,1	30.6
audit-libs	2.8.1	30.6
authconfig	6.2.8	
aws-cfn-bootstrap	2.0	2.0
awscli	1,18,147	
awscli-2		2,15,30
basesystem	10,0	11
bash	4,2,46	5,2,15
bash-completion	2.1	2.11
bc	1,006,95	1,07,1
bind-export-libs	9,11,4	
bind-libs	9,11,4	9,18,28
bind-libs-lite	9,11,4	
bind-license	9,11,4	9,18,28
bind-utils	9,11,4	9,18,28
binutils	2,29,1	2,39
blktrace	1.0.5	
boost-date-time	1,53.0 (x86_64)	
boost-filesystem		1.75,0
boost-system	1,53.0 (x86_64)	1.75,0

Pacchetto	AL2 AMI	AL2023 AMI
boost-thread	1,53.0 (x86_64)	1.75,0
bridge-utils	1.5	
bzip2	1.0.6	1.0.8
bzip2-libs	1.0.6	1.0.8
ca-certificates	2023,2,68	2023,2,68
c-ares		1.19.1
checkpolicy		3.4
chkconfig	1,7,4	1.15
chrony	4.2	4.3
cloud-init	19,3	222,2
cloud-init-cfg-ec2		222,2
cloud-utils-growpart	0,31	0,31
coreutils	8,22	8,32
coreutils-common		8,32
cpio	2,12	2,13
cracklib	2.9.0	2,9,6
cracklib-dicts	2.9.0	29.6
cronie	1,4,11	
cronie-anacron	1,4,11	
crontabs	1.11	1.11

Pacchetto	AL2 AMI	AL2023 AMI
crypto-policies		20220428
crypto-policies-scripts		20220428
cryptsetup	1,7,4	2.6.1
cryptsetup-libs	17.4	2.6.1
curl	8.3.0	
curl-minimal		8,5,0
cyrus-sasl-lib	2,1,26	2,1,27
cyrus-sasl-plain	2,1,26	2,1,27
dbus	1,10,24	1,12,28
dbus-broker		32
dbus-common		1,12,28
dbus-libs	1,10,24	1,12,28
device-mapper	1,02,170	1,02,185
device-mapper-event	1,02,170	
device-mapper-event-libs	1,02,170	
device-mapper-libs	1,02,170	1,02,185
device-mapper-persistent-data	0.7.3	
dhclient	42,5	

Pacchetto	AL2 AMI	AL2023 AMI
dhcp-common	42,5	
dhcp-libs	42,5	
diffutils	3.3	3.8
dmidecode	3.2	
dmraid	1.0.0.rc16	
dmraid-events	1.0.0.rc16	
dnf		4.14.0
dnf-data		4.14.0
dnf-plugin-release-notification		1.2
dnf-plugins-core		4.3.0
dnf-plugin-support-info		1.2
dnf-utils		4.3.0
dosfstools	30,20	4.2
dracut	033	055
dracut-config-ec2	2.0	3.0
dracut-config-generic	033	055
dwz		0,14
dyninst	9.3.1 (x86_64)	10.2.1

Pacchetto	AL2 AMI	AL2023 AMI
e2fsprogs	1,42,9	1,46,5
e2fsprogs-libs	1,42,9	1,46,5
ec2-hibinit-agent	1.0.8	1.0.8
ec2-instance-connect	1.1	1.1
ec2-instance-connect-selinux	1.1	1.1
ec2-net-utils	17.3	
ec2-utils	1.2	2.2.0
ed	1.9	1.14.2
efibootmgr	15 (ogni arco 64)	
efi-filesystem		5
efi-srpm-macros		5
efivar		38
efivar-libs	31 (aarch64)	38
elfutils-debuginfod-client		0.188
elfutils-default-yama-scope	0,176	0.188
elfutils-libelf	0,176	0.188
elfutils-libs	0,176	0.188
ethtool	4,8	5,15

Pacchetto	AL2 AMI	AL2023 AMI
expat	2.1.0	2.5.0
file	5,11	5,39
file-libs	5,11	5,39
filesystem	3.2	3,14
findutils	4,5,11	4.8.0
fipscheck	1.4.1	
fipscheck-lib	1.4.1	
fonts-srpm-macros		2.0.5
freetype	2.8	
fstrm		0.6.1
fuse-libs	29.2	29.9
gawk	40,2	5.1.0
gdbm	1.13	
gdbm-libs		1,19
gdisk	0,8,10	1.0.8
generic-logos	18.0.0	
GeoIP	1.5.0	
gettext	0,198,1	0,21
gettext-libs	0,198,1	0,21
ghc-srpm-macros		1.5.0

Pacchetto	AL2 AMI	AL2023 AMI
glib2	2,56,1	2,74,7
glibc	2,26	2,34
glibc-all-langpacks	2,26	2,34
glibc-common	2,26	2,34
glibc-gconv-extra		2,34
glibc-locale-source	2,26	2,34
glibc-minimal-lang pack	2,26	
gmp	6.0.0	6.2.1
gnupg2	20,22	
gnupg2-minimal		2.3.7
gnutls		3.8.0
go-srpm-macros		3.2.0
gpgme	1.3.2	1.15.1
gpm-libs	1,20,7	1,20.7
grep	2,20	3.8
groff-base	1,22.2	1,22.4
grub2	2,06	
grub2-common	2,06	2,06
grub2-efi-aa64	2,06 (aarch64)	
grub2-efi-aa64-ec2	2.06 (aarch64)	2.06 (aarch64)

Pacchetto	AL2 AMI	AL2023 AMI
grub2-efi-aa64-modules	2.06 (nomarzo)	
grub2-efi-x64-ec2	2.06 (x86_64)	2,06 (x86_64)
grub2-pc	2,06 (x86_64)	
grub2-pc-modules	2.06 (nomarzo)	2.06
grub2-tools	2,06	2,06
grub2-tools-minimal	2,06	2,06
grubby	8,28	8,40
gssproxy	0.7.0	0.8.4
gzip	1.5	1.12
hardlink	1.3	
hibagent	1.1.0	
hostname	3.13	3,23
hunspell	1.3.2	1.7.0
hunspell-en	0,20121024	0,20140811,1
hunspell-en-GB	0,20121024	0,20140811,1
hunspell-en-US	0,20121024	0,20140811,1
hunspell-filesystem		1.7.0
hwdata	0,252	0,384
info	5.1	6.7
inih		49

Pacchetto	AL2 AMI	AL2023 AMI
initscripts	9,49,47	10,09
iproute	5.10.0	6.10.0
iptables	1.8.4	
iptables-libs	1.8.4	
iputils	20180629	20210202
irqbalance	1.7.0	1.9.0
jansson	(2.10)	2.14
jbigkit-libs	2.0	
jemalloc		52,1
jitterentropy		3.4.1
jq		1.7.1
json-c	0,11	0,14
kbd	1,15,5	2.4.0
kbd-legacy	1,15,5	
kbd-misc	1,15,5	2.4.0
kernel	5,10228	6,1112
kernel-libbpf		6,1112
kernel-livepatch-r epo-s3		2023,620241031
kernel-srpm-macros		1.0
kernel-tools	5,10228	6,1112

Pacchetto	AL2 AMI	AL2023 AMI
keyutils	1.5.8	1.6.3
keyutils-libs	1.5.8	1.6.3
kmod	25	29
kmod-libs	25	29
kpartx	0,49	
kpatch-runtime	0.9.4	0,9,7
krb5-libs	1.15.1	1,21,3
langtable	0,0,31	
langtable-data	0,0,31	
langtable-python	0,0,31	
less	458	608
libacl	2,2,51	2.3.1
libaio	0,3109	0,3111
libarchive		3,7,4
libargon2		20171227
libassuan	2.1.0	2,5,5
libattr	2,4,46	2.5.1
libbasicobjects	0,11	01.1
libblkid	2,30,2	2,37,4
libcap	2,54	2,48

Pacchetto	AL2 AMI	AL2023 AMI
libcap-ng	0,7,5	0.8.2
libcbor		0.7.0
libcollection	0.7.0	0.7.0
libcom_err	1,42,9	1,46,5
libcomps		01,20
libconfig	1.4.9	17.2
libcroco	0,6,12	
libcrypt	2,26	
libcurl	8.3,0	
libcurl-minimal		8,5,0
libdaemon	0,14	
libdb	5,3,21	5,3,28
libdb-utils	5,3,21	
libdhash		0,50
libdnf		0,69,0
libdrm	2,4,97	
libdwarf	20130207 (x86_64)	
libeconf		0,4,0
libedit	3.0	3.1
libestr	0,19	

Pacchetto	AL2 AMI	AL2023 AMI
libev		4,33
libevent	2.0,21	2,1,12
libfastjson	0,99,4	
libfdisk	2,30,2	2,37,4
libffi	3,0,13	34.4
libfido2		1.10.0
libgcc	7.3.1	11.4.1
libgcrypt	1.5.3	1.10.2
libgomp	7.3.1	11.4.1
libgpg-error	1.12	1,42
libibverbs		48,0
libicu	50,2	
libidn	1,28	
libidn2	2.3.0	2.3.2
libini_config	1.3.1	1.3.1
libjpeg-turbo	2.0,90	
libkcapi		1.4.0
libkcapi-hmaccalc		1.4.0
libldb		26.2
libmaxminddb		1.5.2

Pacchetto	AL2 AMI	AL2023 AMI
libmetalink	0,13	0,13
libmnl	1.0.3	1.0.4
libmodulemd		2.13.0
libmount	2,30,2	2,37,4
libnetfilter_connt rack	1.0.6	
libnfnetworking	1.0.1	
libnfsidmap	0.25	2,5,4
libnghttp2	1,41,0	1,59,0
libnl3	3,2,28	3.5.0
libnl3-cli	3,2,28	
libpath_utils	0,21	0,2,1
libpcap	1.5.3	1.10.1
libpciaccess	0,14 (x86_64)	
libpipeline	1.2.3	1.5.3
libpkgconf		1.8.0
libpng	1.5.13	
libpsl	0,21,5	0,21,1
libpwquality	1.2.3	1.4.4
libref_array	0,1,5	0,1,5
librepo		1,14,5

Pacchetto	AL2 AMI	AL2023 AMI
libreport-filesystem		2,15,2
libseccomp	2.5.2	2.5.3
libselinux	2.5	3.4
libselinux-utils	2.5	3.4
libsemanage	2.5	3.4
libsepol	2.5	3.4
libsigsegv		2,13
libsmartcols	2,30,2	2,37,4
libsolv		0,7,22
libss	1,42,9	1,46,5
libssh2	1.4.3	
libsss_certmap		2,9,4
libsss_idmap	1,16,5	2,9,4
libsss_nss_idmap	1,16,5	2,9,4
libsss_sudo		2,9,4
libstdc++	7.3.1	11.4.1
libstoragemgmt	1.6.1	1.9.4
libstoragemgmt-python	1.6.1	
libstoragemgmt-python-clibs	1.6.1	

Pacchetto	AL2 AMI	AL2023 AMI
libsysfs	2.1.0	
libtalloc		2.3.4
libtasn1	4,10	4,19,0
libtdb		1.4.7
libteam	1,27	
libtevent		0.13.0
libtextstyle		0,21
libtiff	40,3	
libtirpc	02.4	1.3.3
libunistring	0,9,3	0,9,10
libuser	0,60	0,63
libutempter	1.1.6	1.2.1
libuuid	2,30,2	2,37,4
libuv		1.47.0
libverto	02,5	0,32
libverto-libev		0,32
libverto-libevent	02,5	
libwebp	0,3,0	
libxcrypt		4,4,33
libxml2	2,9,1	210.4

Pacchetto	AL2 AMI	AL2023 AMI
libxml2-python	29.1	
libyaml	0,14	02,5
libzstd		1,5,5
linux-firmware-whe nce		20210208 (novembre)
lm_sensors-libs	3.4.0	3.6.0
lmdb-libs		0,9,29
logrotate	3,86	3,20,1
lsof	4,87	4,94,0
lua	5.1.4	
lua-libs		5.4.4
lua-srpm-macros		1
lvm2	2,02,187	
lvm2-libs	2,02,187	
lz4	1,7,5	
lz4-libs		1.9.4
make	3,82	
man-db	26.3	29.3
man-pages	3,53	5,10
man-pages-overrides	7,5,2	
mariadb-libs	5,5,68	

Pacchetto	AL2 AMI	AL2023 AMI
mdadm	4.0	
microcode_ctl	2.1 (x86_64)	2.1 (x86_64)
mlocate	0,26	
mpfr		4.1.0
mtr	0.92	
nano	2,9,8	5.8
ncurses	6.0	6.2
ncurses-base	6.0	6.2
ncurses-libs	6.0	6.2
nettle	2.7.1	3.8
net-tools	2.0	2.0
newt	0,52,15	0,52,21
newt-python	0,52,15	
nfs-utils	1.3.0	2,5,4
npth		1.6
nspr	4,35,0	4,35,0
nss	3,90,0	3,90,0
nss-pem	1.0.3	
nss-softokn	3,90,0	3,90,0
nss-softokn-freebl	3,90,0	3,90,0

Pacchetto	AL2 AMI	AL2023 AMI
nss-sysinit	3,90,0	3,90,0
nss-tools	3,90,0	
nss-util	3,90,0	3,90,0
ntsysv	17.4	1.15
numactl-libs	2.0.9	2.0,14
ocaml-srpm-macros		6
oniguruma		6,9,7,1
openblas-srpm-macros		2
openldap	2,4,44	2,4,57
openssh	7,4p 1	8,7p1
openssh-clients	7,4p1	8,7p1
openssh-server	7,4p1	8,7p1
openssl	1,0,2k	3,0,8
openssl-libs	1,0,2k	3,0,8
openssl-pkcs11		0,4,12
os-prober	1.58	1,77
p11-kit	0,23,22	0,24,1
p11-kit-trust	0,23,22	0,24,1
package-notes-srpm-macros		0.4
pam	1.1.8	1.5.1

Pacchetto	AL2 AMI	AL2023 AMI
parted	3.1	3.4
passwd	0,79	0,80
pciutils	35,1	3.7.0
pciutils-libs	35.1	3.7.0
pcre	8,32	
pcre2	10,23	10,40
pcre2-syntax		10,40
perl	5,16,3	
perl-Carp	1,26	1,50
perl-Class-Struct		0,66
perl-constant	1,27	1,33
perl-DynaLoader		1,47
perl-Encode	2,51	3,15
perl-Errno		1,30
perl-Exporter	5,68	5,74
perl-Fcntl		1.13
perl-File-Basename		2,85
perl-File-Path	2,09	2,18
perl-File-stat		1,09
perl-File-Temp	0,23,01	0,231,100

Pacchetto	AL2 AMI	AL2023 AMI
perl-Filter	1,49	
perl-Getopt-Long	2,40	2,52
perl-Getopt-Std		1.12
perl-HTTP-Tiny	0,033	0,078
perl-if		0,60,800
perl-interpreter		5,32,1
perl-IO		1,43
perl-IPC-Open3		1,21
perl-libs	5,16,3	5,32,1
perl-macros	5,16,3	
perl-MIME-Base64		3,16
perl-mro		1,23
perl-overload		1,31
perl-overloading		0,02
perl-parent	0,225	0,238
perl-PathTools	3,40	3,78
perl-Pod-Escapes	1.04	1,07
perl-podlators	2.5.1	4,14
perl-Pod-Perldoc	3,20	3,28,01
perl-Pod-Simple	3,28	3,42

Pacchetto	AL2 AMI	AL2023 AMI
perl-Pod-Usage	1,63	2,01
perl-POSIX		1,94
perl-Scalar-List-Utils	1,27	1,56
perl-SelectSaver		1.02
perl-Socket	2,010	2,032
perl-srpm-macros		1
perl-Storable	2,45	3,21
perl-subst		1,03
perl-Symbol		1,08
perl-Term-ANSIColor		5,01
perl-Term-Cap		1,17
perl-Text-ParseWords	3,29	3,30
perl-Text-Tabs+Wrap		2021,0726
perl-threads	1,87	
perl-threads-shared	1,43	
perl-Time-HiRes	1,9725	
perl-Time-Local	1,2300	1.300
perl-vars		1,05
pinentry	0.8.1	
pkgconf		1.8.0

Pacchetto	AL2 AMI	AL2023 AMI
pkgconfig	0,27,1	
pkgconf-m4		1.8.0
pkgconf-pkg-config		1.8.0
plymouth	0,89	
plymouth-core-libs	0,89	
plymouth-scripts	0,89	
pm-utils	1.4.1	
policycoreutils	2.5	3.4
policycoreutils-python-utils		3.4
popt	1.13	1,18
postfix	210.1	
procps-ng	3,3,10	3,3,17
protobuf-c		1.4.1
psacct	6.6.1	6.6.4
psmisc	22,20	23,4
pth	2.0.7	
publicsuffix-list-dafsa	20240208	20240212
pygpgme	0.3	
pyliblzma	0,53	

Pacchetto	AL2 AMI	AL2023 AMI
pystache	0,5,3	
python	2,7,18	
python2-botocore	1,18,6	
python2-colorama	0,39	
python2-cryptography	17.2	
python2-dateutil	2.6.1	
python2-futures	3,0,5	
python2-jmespath	0,9,3	
python2-jsonschema	2.5.1	
python2-oauthlib	2.0.1	
python2-pyasn1	0,19	
python2-rpm	4.11.3	
python2-rsa	3.4.1	
python2-s3transfer	0,3,3	
python2-setuptools	41,2,0	
python2-six	1.11.0	
python3	3,7,16	3,9,16
python3-attrs		20,3,0
python3-audit		30.6
python3-awscrt		0,19,19

Pacchetto	AL2 AMI	AL2023 AMI
python3-babel		2,9,1
python3-cffi		1,14,5
python3-chardet		4.0.0
python3-colorama		04.4
python3-configobj		50.6
python3-cryptography		36,0
python3-daemon	2.2.3	2.3.0
python3-dateutil		28.1
python3-dbus		1,2,18
python3-distro		1.5.0
python3-dnf		4.14.0
python3-dnf-plugins-core		4.3.0
python3-docutils	0,14	0,16
python3-gpg		1.15.1
python3-hawkey		0,69,0
python3-idna		(2.10)
python3-jinja2		211,3
python3-jmespath		0.10.0
python3-jsonpatch		1,21
python3-jsonpointer		2.0

Pacchetto	AL2 AMI	AL2023 AMI
python3-jsonschema		3.2.0
python3-libcomps		01,20
python3-libdnf		0,69,0
python3-libs	3,7,16	3,9,16
python3-libselinux		3.4
python3-libsemanage		3.4
python3-libstorage mgmt		1.9.4
python3-lockfile	0.11.0	0.12.2
python3-markupsafe		1.1.1
python3-netifaces		0,10,6
python3-oauthlib		3.0.2
python3-pip	202,2	
python3-pip-wheel		21,31
python3-ply		3,11
python3-policycore utils		3.4
python3-prettytable		0.7.2
python3-prompt-too lkit		3,0,24
python3-pycparser		2,20

Pacchetto	AL2 AMI	AL2023 AMI
python3-pyrsistent		0,17,3
python3-pyserial		3.4
python3-pysocks		1.7.1
python3-pystache	0,5,4	
python3-pytz		2022,7,1
python3-pyyaml		5.4.1
python3-requests		2,25,1
python3-rpm		4,161,3
python3-ruamel-yaml		0,16,6
python3-ruamel-yaml-clib		0,12
python3-setools		4.4.1
python3-setuptools	49,13	59,60
python3-setuptools-wheel		59,6,0
python3-simplejson	3.2.0	
python3-six		1.15.0
python3-systemd		235
python3-urllib3		1,25,10
python3-wcwidth		0,2,5
python-babel	0.9.6	

Pacchetto	AL2 AMI	AL2023 AMI
python-backports	1.0	
python-backports-s sl_match_hostname	3,50,1	
python-cffi	1.6.0	
python-chardet	2.2.1	
python-chevron		0.13.1
python-configobj	4,7,2	
python-daemon	1.6	
python-devel	2,7,18	
python-docutils	0,12	
python-enum34	1.0.4	
python-idna	2.4	
python-iniparse	0.4	
python-ipaddress	1,0,16	
python-jinja2	2,7,2	
python-jsonpatch	1.2	
python-jsonpointer	1.9	
python-jwcrypto	0,4,2	
python-kitchen	1.1.1	
python-libs	2,7,18	
python-lockfile	0.9.1	

Pacchetto	AL2 AMI	AL2023 AMI
python-markupsafe	0,11	
python-pillow	2.0.0	
python-ply	3.4	
python-pycparser	2.14	
python-pycurl	7,19,0	
python-repoze-lru	0.4	
python-requests	2.6.0	
python-simplejson	3.2.0	
python-srpm-macros		3.9
python-urlgrabber	3,10	
python-urllib3	1,25,9	
pyxattr	0,5,1	
PyYAML	3,10	
qrencode-libs	3.4.1	
quota	4,01	4,06
quota-nls	4,01	4,06
rdate	1.4	
readline	6.2	8.1
rng-tools	6.8	6,14
rootfiles	8.1	8.1

Pacchetto	AL2 AMI	AL2023 AMI
rpcbind	0,2,0	1.2.6
rpm	4.11.3	4,161,3
rpm-build-libs	411,3	4,161,3
rpm-libs	411,3	4,161,3
rpm-plugin-selinux		4,161,3
rpm-plugin-systemd-inhibit	411,3	4,161,3
rpm-sign-libs		4,161,3
rsync	3.1.2	3.2.6
rsyslog	8,24,0	
rust-srpm-macros		21
sbsigntools		0.9.4
scl-utils	20130529	
screen	4.1.0	4.8.0
sed	42,2	4.8
selinux-policy	3,13,1	381,45
selinux-policy-targeted	3,13,1	381,45
setserial	2,17	
setup	2,8,71	2,13,7
setuptools	1,19,11	

Pacchetto	AL2 AMI	AL2023 AMI
sgpio	1,2,0,10	
shadow-utils	4.1.5.1	4.9
shared-mime-info	1.8	
slang	2.2.4	2.3.2
sqlite	3,7,17	
sqlite-libs		3,4,0
sssd-client	1,16,5	2,9,4
sssd-common		2,9,4
sssd-kcm		2,9,4
sssd-nfs-idmap		2,9,4
strace	4,26	6.8
sudo	1,8,23	1,9,15
sysctl-defaults	1.0	1.0
sysstat	101,5	12,5,6
systemd	219	252,23
systemd-libs	219	252,23
systemd-networkd		252,23
systemd-pam		252,23
systemd-resolved		252,23
systemd-sysv	219	

Pacchetto	AL2 AMI	AL2023 AMI
systemd-udev		252,23
system-release	2	2023,620241031
systemtap-runtime	4,5	4.8
sysvinit-tools	2,88	
tar	1,26	1,34
tbb		2020,3
tcp_wrappers	7.6	
tcp_wrappers-libs	7.6	
tcpdump	4,9,2	4,99,1
tcsh	6,18,01	6,24,07
teamd	1,27	
time	1,7	1.9
traceroute	20,22	2.1.3
tzdata	2024a	2024a
unzip	6.0	6.0
update-motd	1.1.2	2.2
usermode	1,111	
userspace-rcu		0.12.1
ustr	1.0.4	
util-linux	2,30,2	2,37,4

Pacchetto	AL2 AMI	AL2023 AMI
util-linux-core		2,37,4
vim-common	9,0,2153	9,0,2153
vim-data	9,0,2153	9,0,2153
vim-enhanced	9,0,2153	9,0,2153
vim-filesystem	9,0,2153	9,0,2153
vim-minimal	9,0,2153	9,0,2153
virt-what	1,18	
wget	1.14	1,21,3
which	2,20	2,21
words	3.0	3.0
xfsdump	3,18	3,1,11
xfspgms	5.0.0	5,18,0
xxd	9,0,2153	9,0,2153
xxhash-libs		0.8.0
xz	5.2.2	5.2.5
xz-libs	5.2.2	5.2.5
yajl	2.0.4	
yum	3.4.3	4.14.0
yum-langpacks	04.2	
yum-metadata-parser	1.1.4	

Pacchetto	AL2 AMI	AL2023 AMI
yum-plugin-priorities	1,1,31	
yum-utils	1,1,31	
zip	3.0	3.0
zlib	1.2.7	1,2,11
zram-generator		1.1.2
zram-generator-defaults		1.1.2
zstd		1,5,5

Confronto dei pacchetti installati su Amazon Linux 2 e Amazon Linux 2023 Minimal AMIs

Un confronto tra il RPMs presente su Amazon Linux 2 e AL2 023 Minimal AMIs.

Pacchetto	AL2 Minimo	AL2023 Minimo
acl	2.2.51	
alternatives		1.15
amazon-chrony-config		4.3
amazon-ec2-net-utils		2.5.1
amazon-linux-extras	2.0.3	
amazon-linux-repo-s3		2023,620241031
amazon-linux-sb-keys		2023,1

Pacchetto	AL2 Minimo	AL2023 Minimo
amd-ucode-firmware	21 aprile 2020 (marzo)	20210208 (nomarzo)
audit	2.8.1	30.6
audit-libs	2.8.1	30.6
authconfig	6.2.8	
awscli-2		2,15,30
basesystem	10,0	11
bash	4,2,46	5,2,15
bind-export-libs	9,11,4	
bzip2-libs	1.0.6	1.0.8
ca-certificates	2023,2,68	2023,2,68
checkpolicy		3.4
chkconfig	1,7,4	
chrony	4.2	4.3
cloud-init	19,3	222,2
cloud-init-cfg-ec2		222,2
cloud-utils-growpart	0,31	0,31
coreutils	8,22	8,32
coreutils-common		8,32
cpio	2,12	2,13
cracklib	2.9.0	2,9,6

Pacchetto	AL2 Minimo	AL2023 Minimo
cracklib-dicts	2.9.0	29.6
cronie	1,4,11	
cronie-anacron	1,4,11	
crontabs	1.11	
crypto-policies		20220428
cryptsetup-libs	1,7,4	2.6.1
curl	8.3.0	
curl-minimal		8,5,0
cyrus-sasl-lib	2,1,26	2,1,27
dbus	1,10,24	1,12,28
dbus-broker		32
dbus-common		1,12,28
dbus-libs	1,10,24	1,12,28
device-mapper	1,02,170	1,02,185
device-mapper-libs	1,02,170	1,02,185
dhclient	42,5	
dhcp-common	42,5	
dhcp-libs	42,5	
diffutils	3.3	3.8
dnf		4.14.0

Pacchetto	AL2 Minimo	AL2023 Minimo
dnf-data		4.14.0
dnf-plugin-release-notification		1.2
dnf-plugins-core		4.3.0
dnf-plugin-support-info		1.2
dracut	033	055
dracut-config-ec2	2.0	3.0
dracut-config-generic	033	055
e2fsprogs	1,42,9	1,46,5
e2fsprogs-libs	1,42,9	1,46,5
ec2-utils	1.2	2.2.0
efibootmgr	15 (ogni arco 64)	
efi-filesystem		5
efivar		38
efivar-libs	31 (aarch64)	38
elfutils-default-yama-scope	0,176	0.188
elfutils-libelf	0,176	0.188
elfutils-libs	0,176	0.188
expat	2.1.0	2.5.0

Pacchetto	AL2 Minimo	AL2023 Minimo
file	5,11	5,39
file-libs	5,11	5,39
filesystem	3.2	3,14
findutils	4,5,11	4.8.0
fipscheck	1.4.1	
fipscheck-lib	1.4.1	
freetype	2.8	
fuse-libs	29.2	29.9
gawk	40,2	5.1.0
gdbm	1.13	
gdbm-libs		1,19
gdisk	0,8,10	1.0.8
gettext	0,198,1	0,21
gettext-libs	0,198,1	0,21
glib2	2,56,1	2,74,7
glibc	2,26	2,34
glibc-all-langpacks	2,26	2,34
glibc-common	2,26	2,34
glibc-locale-source	2,26	2,34
glibc-minimal-lang pack	2,26	

Pacchetto	AL2 Minimo	AL2023 Minimo
gmp	6.0.0	6.2.1
gnupg2	20,22	
gnupg2-minimal		2.3.7
gnutls		3.8.0
gpgme	1.3.2	1.15.1
grep	2,20	3.8
groff-base	1,22.2	1,22.4
grub2	2,06	
grub2-common	2,06	2,06
grub2-efi-aa64	2,06 (aarch64)	
grub2-efi-aa64-ec2	2.06 (aarch64)	2.06 (aarch64)
grub2-efi-aa64-modules	2.06 (nomarzo)	
grub2-efi-x64-ec2	2.06 (x86_64)	2,06 (x86_64)
grub2-pc	2,06 (x86_64)	
grub2-pc-modules	2.06 (nomarzo)	2.06
grub2-tools	2,06	2,06
grub2-tools-minimal	2,06	2,06
grubby	8,28	8,40
gzip	1.5	1.12
hardlink	1.3	

Pacchetto	AL2 Minimo	AL2023 Minimo
hostname	3.13	3,23
hwdata		0,384
info	5.1	
inih		49
initscripts	9,49,47	10,09
iproute	5.10.0	6.10.0
iptables	1.8.4	
iptables-libs	1.8.4	
iputils	20180629	20210202
irqbalance	1.7.0	1.9.0
jansson		2.14
jitterentropy		3.4.1
jq		1.7.1
json-c		0,14
kbd		2.4.0
kbd-misc		2.4.0
kernel	4,14,355	6,1112
kernel-libbpf		6,1112
kernel-livepatch-r epo-s3		2023,620241031
keyutils-libs	1,58	1.6.3

Pacchetto	AL2 Minimo	AL2023 Minimo
kmod	25	29
kmod-libs	25	29
kpartx	0,49	
krb5-libs	1.15.1	1,21,3
less	458	608
libacl	2,2,51	2.3.1
libarchive		3,7,4
libargon2		20171227
libassuan	2.1.0	2,5,5
libattr	2,4,46	2.5.1
libblkid	2,30,2	2,37,4
libcap	2,54	2,48
libcap-ng	0,7,5	0.8.2
libcbor		0.7.0
libcom_err	1,42,9	1,46,5
libcomps		01,20
libcroco	0,6,12	
libcrypt	2,26	
libcurl	8.3,0	
libcurl-minimal		8,5,0

Pacchetto	AL2 Minimo	AL2023 Minimo
libdb	5,3,21	5,3,28
libdb-utils	5,3,21	
libdnf		0,69,0
libeconf		0,40
libedit	3.0	3.1
libestr	0,19	
libfastjson	0,99,4	
libfdisk	2,30,2	2,37,4
libffi	3,0,13	34.4
libfido2		1.10.0
libgcc	7.3.1	114,1
libgcrypt	1.5.3	1.10.2
libgomp	7.3.1	11.4.1
libgpg-error	1.12	1,42
libicu	50,2	
libidn	1,28	
libidn2	2.3.0	2.3.2
libkcapi		1.4.0
libkcapi-hmacalc		1.4.0
libmetalink	0,13	

Pacchetto	AL2 Minimo	AL2023 Minimo
libmnl	1.0.3	1.0.4
libmodulemd		2.13.0
libmount	2,30,2	2,37,4
libnetfilter_conntrack	1.0.6	
libnfnetlink	1.0.1	
libnhttp2	1,41,0	1,59,0
libpcap	1.5.3	
libpipeline	1.2.3	1.5.3
libpng	1,5,13	
libpsl	0,21,5	0,21,1
libpwquality	1.2.3	1.4.4
librepo		1,14,5
libreport-filessystem		2,15,2
libseccomp	2.5.2	2.5.3
libselinux	2.5	3.4
libselinux-utils	2.5	3.4
libsemanage	2.5	3.4
libsepol	2.5	3.4
libsigsegv		2,13
libsmartcols	2,30,2	2,37,4

Pacchetto	AL2 Minimo	AL2023 Minimo
libsolv		0,7,22
libss	1,42,9	1,46,5
libssh2	1.4.3	
libstdc++	7.3.1	11.4.1
libsysfs	2.1.0	
libtasn1	4,10	4,19,0
libtextstyle		0,21
libunistring	0,9,3	0,9,10
libuser	0,60	0,63
libutempter	1.1.6	1.2.1
libuuid	2,30,2	2,37,4
libverto	0,2,5	0,32
libxcrypt		4,4,33
libxml2	2,9,1	210.4
libyaml	0,14	02,5
libzstd		1,5,5
linux-firmware-whe nce		20210208 (novembre)
logrotate	3.8.6	3,20,1
lua	5.1.4	
lua-libs		5.4.4

Pacchetto	AL2 Minimo	AL2023 Minimo
lz4	1,7,5	
lz4-libs		1.9.4
make	3,82	
man-db	26.3	29.3
mariadb-libs	5,5,68	
microcode_ctl	2.1 (x86_64)	2.1 (x86_64)
mpfr		4.1.0
ncurses	6.0	6.2
ncurses-base	6.0	6.2
ncurses-libs	6.0	6.2
nettle	2.7.1	3.8
net-tools	2.0	2.0
newt	0,52,15	
newt-python	0,52,15	
npth		1.6
nspr	4,35,0	
nss	3,90,0	
nss-pem	1.0.3	
nss-softokn	3,90,0	
nss-softokn-freebl	3,90,0	

Pacchetto	AL2 Minimo	AL2023 Minimo
nss-sysinit	3,90,0	
nss-tools	3,90,0	
nss-util	3,90,0	
numactl-libs	20,9	2.0,14
oniguruma		6,9,7,1
openldap	2,4,44	2,4,57
openssh	7,4p 1	8,7p1
openssh-clients	7,4p1	8,7p1
openssh-server	7,4p1	8,7p1
openssl	1,0,2k	3,0,8
openssl-libs	1,0,2k	3,0,8
openssl-pkcs11		0,4,12
os-prober	1.58	1,77
p11-kit	0,23,22	0,24,1
p11-kit-trust	0,23,22	0,24,1
pam	1.1.8	1.5.1
passwd	0,79	0,80
pciutils		3.7.0
pciutils-libs		3.7.0
pcre	8,32	

Pacchetto	AL2 Minimo	AL2023 Minimo
pcr2	10,23	10,40
pcr2-syntax		10,40
pinentry	0.8.1	
pkgconfig	0,27,1	
policycoreutils	2.5	3.4
popt	1.13	1,18
postfix	210.1	
procps-ng	3,3,10	3,3,17
psmisc	22,20	23,4
pth	2.0.7	
publicsuffix-list-dafsa	20240208	20240212
pygpgme	0.3	
pyliblzma	0,53	
python	2,7,18	
python2-cryptography	17.2	
python2-jsonschema	2.5.1	
python2-oauthlib	2.0.1	
python2-pyasn1	0,19	
python2-rpm	4.11.3	
python2-setuptools	41,2,0	

Pacchetto	AL2 Minimo	AL2023 Minimo
python2-six	1.11.0	
python3		3,9,16
python3-attrs		20,3,0
python3-audit		30.6
python3-awscrt		0,19,19
python3-babel		2,9,1
python3-cffi		1,14,5
python3-chardet		4.0.0
python3-colorama		04.4
python3-configobj		5.0.6
python3-cryptography		36,0
python3-dateutil		28.1
python3-dbus		1,2,18
python3-distro		1.5.0
python3-dnf		4.14.0
python3-dnf-plugins-core		4.3.0
python3-docutils		0,16
python3-gpg		1.15.1
python3-hawkey		0,69,0
python3-idna		(2.10)

Pacchetto	AL2 Minimo	AL2023 Minimo
python3-jinja2		211,3
python3-jmespath		0.10.0
python3-jsonpatch		1,21
python3-jsonpointer		2.0
python3-jsonschema		3.2.0
python3-libcomps		01,20
python3-libdnf		0,69,0
python3-libs		3,9,16
python3-libseltlinux		3.4
python3-libsemanage		3.4
python3-markupsafe		1.1.1
python3-netifaces		0,10,6
python3-oauthlib		3.0.2
python3-pip-wheel		21,31
python3-ply		3,11
python3-policycore utils		3.4
python3-prettytable		0.7.2
python3-prompt-too lkit		3,0,24
python3-pycparser		2,20

Pacchetto	AL2 Minimo	AL2023 Minimo
python3-pyrsistent		0,17,3
python3-pyserial		3.4
python3-pysocks		1.7.1
python3-pytz		2022,7,1
python3-pyyaml		5.4.1
python3-requests		2,25,1
python3-rpm		4,161,3
python3-ruamel-yaml		0,16,6
python3-ruamel-yaml-clib		0,12
python3-setools		4.4.1
python3-setuptools		59,6,0
python3-setuptools-wheel		59,6,0
python3-six		1.15.0
python3-systemd		235
python3-urllib3		1,25,10
python3-wcwidth		0,2,5
python-babel	0.9.6	
python-backports	1.0	

Pacchetto	AL2 Minimo	AL2023 Minimo
python-backports-s sl_match_hostname	3,50,1	
python-cffi	1.6.0	
python-chardet	2.2.1	
python-configobj	4,7,2	
python-devel	2,7,18	
python-enum34	1.0.4	
python-idna	2.4	
python-iniparse	0.4	
python-ipaddress	1,0,16	
python-jinja2	2,7,2	
python-jsonpatch	1.2	
python-jsonpointer	1.9	
python-jwcrypto	0,4,2	
python-libs	2,7,18	
python-markupsafe	0,11	
python-ply	3.4	
python-pycparser	2.14	
python-pycurl	7,19,0	
python-repoze-lru	0.4	
python-requests	2.6.0	

Pacchetto	AL2 Minimo	AL2023 Minimo
python-urlgrabber	3,10	
python-urllib3	1,25,9	
pyxattr	0,5,1	
PyYAML	3,10	
qrencode-libs	3.4.1	
readline	6.2	8.1
rng-tools	6.8	6,14
rootfiles	8.1	8.1
rpm	4.11.3	4,161,3
rpm-build-libs	411,3	4,161,3
rpm-libs	4.11,3	4,161,3
rpm-plugin-selinux		4,161,3
rpm-plugin-systemd-inhibit	4.11,3	4,161,3
rpm-sign-libs		4,161,3
rsyslog	8,24,0	
sbsigntools		0.9.4
sed	4.2.2	4.8
selinux-policy	3,13,1	381,45
selinux-policy-targeted	3,13,1	381,45

Pacchetto	AL2 Minimo	AL2023 Minimo
setup	2,8,71	2,13,7
shadow-utils	4.1.5.1	4.9
shared-mime-info	1.8	
slang	2.2.4	
sqlite	3,7,17	
sqlite-libs		3,4,0
sudo	1,8,23	1,9,15
sysctl-defaults	1.0	1.0
systemd	219	252,23
systemd-libs	219	252,23
systemd-networkd		252,23
systemd-pam		252,23
systemd-resolved		252,23
systemd-sysv	219	
systemd-udev		252,23
system-release	2	2023,620241031
sysvinit-tools	2,88	
tar	1,26	1,34
tcp_wrappers-libs	7.6	
tzdata	2024a	2024a

Pacchetto	AL2 Minimo	AL2023 Minimo
update-motd	1.1.2	2.2
userspace-rcu		0.12.1
ustr	1.0.4	
util-linux	2,30,2	2,37,4
util-linux-core		2,37,4
vim-data	9,0,2153	9,0,2153
vim-minimal	9,0,2153	9,0,2153
which	2,20	2,21
xfspgrog	5.0.0	5,18,0
xz	5.2.2	5.2.5
xz-libs	5.2.2	5.2.5
yum	3.4.3	4.14.0
yum-metadata-parser	1.1.4	
yum-plugin-priorities	1,1,31	
zlib	1.2.7	1,2,11
zram-generator		1.1.2
zram-generator-defaults		1.1.2
zstd		1,5,5

Confronto dei pacchetti installati sulle immagini dei container di base Amazon Linux 2 e Amazon Linux 2023

Un confronto tra le immagini RPMs presenti sui container di base Amazon Linux 2 e AL2 023.

Pacchetto	AL2 Contenitore	AL2Contenitore 023
alternatives		1.15
amazon-linux-extras	2.0.3	
amazon-linux-repo-cdn		2023,620241031
audit-libs		3,0,6
basesystem	10,0	11
bash	4,2,46	5,2,15
bzip2-libs	1.0.6	1.0.8
ca-certificates	2023,2,68	2023,2,68
chkconfig	1,7,4	
coreutils	8,22	
coreutils-single		8,32
cpio	2,12	
crypto-policies		20220428
curl	8,3,0	
curl-minimal		8,5,0
cyrus-sasl-lib	2,1,26	

Pacchetto	AL2 Contenitore	AL2Contenitore 023
diffutils	3.3	
dnf		4.14.0
dnf-data		4.14.0
elfutils-default-yama-scope		0.188
elfutils-libelf	0,176	0.188
elfutils-libs		0.188
expat	2.1.0	2.5.0
file-libs	5,11	5,39
filesystem	3.2	3,14
findutils	4,5,11	
gawk	40,2	5.1.0
gdbm	1.13	
gdbm-libs		1,19
glib2	2,56,1	2,74,7
glibc	2,26	2,34
glibc-common	2,26	2,34
glibc-langpack-en	2,26	
glibc-minimal-langpack	2,26	2,34
gmp	6.0.0	62,1

Pacchetto	AL2 Contenitore	AL2Contenitore 023
gnupg2	20,22	
gnupg2-minimal		2.3.7
gpgme	1.3.2	1.15.1
grep	2,20	3.8
info	5.1	
json-c		0,14
keyutils-libs	1.5.8	1.6.3
krb5-libs	1.15.1	1,21,3
libacl	2,2,51	2.3.1
libarchive		3,7,4
libassuan	2.1.0	2,5,5
libattr	2,4,46	2.5.1
libblkid	2,30,2	2,37,4
libcap	2,54	2,48
libcap-ng		0.8.2
libcom_err	1,42,9	1,46,5
libcomps		01,20
libcrypt	2,26	
libcurl	8.3,0	
libcurl-minimal		8,5,0

Pacchetto	AL2 Contenitore	AL2Contenitore 023
libdb	5,3,21	
libdb-utils	5,3,21	
libdnf		0,69,0
libffi	3,0,13	34.4
libgcc	7.3.1	114,1
libgcrypt	1.5.3	1.10.2
libgomp		11.4.1
libgpg-error	1.12	1,42
libidn2	2.3.0	2.3.2
libmetalink	0,13	
libmodulemd		2.13.0
libmount	2,30,2	2,37,4
libnghttp2	1,41,0	1,59,0
libpsl	0,21,5	0,21,1
librepo		1,14,5
libreport-filessystem		2,15,2
libselenium	2.5	3.4
libsepol	2.5	3.4
libsigsegv		2,13
libsmartcols		2,37,4

Pacchetto	AL2 Contenitore	AL2Contenitore 023
libsolv		0,7,22
libssh2	1.4.3	
libstdc++	7.3.1	11,4,1
libtasn1	4,10	4,19,0
libunistring	0,9,3	0,9,10
libuuid	2,30,2	2,37,4
libverto	0,2,5	0,32
libxcrypt		4,4,33
libxml2	2,9,1	210.4
libyaml		02,5
libzstd		1,5,5
lua	5.1.4	
lua-libs		5.4.4
lz4-libs		1.9.4
mpfr		4.1.0
ncurses	6.0	
ncurses-base	6.0	6.2
ncurses-libs	6.0	6.2
npth		1.6
nspr	4,35,0	

Pacchetto	AL2 Contenitore	AL2Contenitore 023
nss	3,90,0	
nss-pem	1.0.3	
nss-softokn	3,90,0	
nss-softokn-freebl	3,90,0	
nss-sysinit	3,90,0	
nss-tools	3,90,0	
nss-util	3,90,0	
openldap	2,4,44	
openssl-lib	1,0,2k	3,0,8
p11-kit	0,23,22	0,24,1
p11-kit-trust	0,23,22	0,24,1
pcre	8,32	
pcre2		10,40
pcre2-syntax		10,40
pinentry	0.8.1	
popt	1.13	1,18
pth	2.0.7	
publicsuffix-list-dafsa	20240208	20240212
pygpgme	0.3	
pyliblzma	0,53	

Pacchetto	AL2 Contenitore	AL2Contenitore 023
python	2,7,18	
python2-rpm	411,3	
python3		3,9,16
python3-dnf		4.14.0
python3-gpg		1.15.1
python3-hawkey		0,69,0
python3-libcomps		01,20
python3-libdnf		0,69,0
python3-libs		3,9,16
python3-pip-wheel		21,31
python3-rpm		4,16,1,3
python3-setuptools-wheel		59,60
python-iniparse	0.4	
python-libs	2,7,18	
python-pycurl	7,19,0	
python-urlgrabber	3,10	
pyxattr	0,5,1	
readline	6.2	8.1
rpm	4.11.3	4,161,3
rpm-build-libs	411,3	4,161,3

Pacchetto	AL2 Contenitore	AL2Contenitore 023
rpm-libs	411,3	4,161,3
rpm-sign-libs		4,161,3
sed	4.2.2	4.8
setup	2,8,71	2,13,7
shared-mime-info	1.8	
sqlite	3,7,17	
sqlite-libs		3,4,0
system-release	2	2023,620241031
tzdata	2024a	2024a
vim-data	9,0,2153	
vim-minimal	9,0,2153	
xz-libs	5.2.2	5.2.5
yum	3.4.3	4.14.0
yum-metadata-parser	1.1.4	
yum-plugin-ovl	1,1,31	
yum-plugin-priorities	1,1,31	
zlib	1.2.7	1,2,11

Confronto AL1 e AL2 023

[I seguenti argomenti descrivono le differenze principali tra AL1 e AL2 023 che non sono già state incluse nel confronto con. AL2](#)

Note

AL1 ha raggiunto il suo end-of-life (EOL) il 31 dicembre 2023 e non riceverà aggiornamenti di sicurezza o correzioni di bug a partire dal 1° gennaio 2024. Per ulteriori informazioni su AL1 EOL e supporto di manutenzione, consulta il post del blog [Update on Amazon Linux AMI end-of-life](#). Ti consigliamo di aggiornare le applicazioni alla versione AL2 023, che include il supporto a lungo termine fino al 2028.

Argomenti

- [Supporto per ogni rilascio](#)
- [systemd sostituisce upstart come sistema init](#)
- [Python 2.6 e 2.7 sono stati sostituiti con Python 3](#)
- [OpenJDK 8 come JDK più vecchio](#)
- [AL2023 modifiche al kernel rispetto ad Amazon Linux 1 \(\) AL1](#)
- [Confronto dei pacchetti installati su Amazon Linux 1 \(AL1\) e Amazon Linux 2023 AMIs](#)
- [Confronto dei pacchetti installati su Amazon Linux 1 \(AL1\) e Amazon Linux 2023 Minimal AMIs](#)
- [Confronto dei pacchetti installati sulle immagini dei container di base Amazon Linux 1 \(AL1\) e Amazon Linux 2023](#)

Supporto per ogni rilascio

Per AL2 023, offriamo cinque anni di supporto dalla data di rilascio. AL1 il supporto standard è terminato il 31 dicembre 2020 e il supporto di manutenzione è terminato il 31 dicembre 2023.

Per ulteriori informazioni, consulta [Cadenza di rilascio](#).

systemd sostituisce upstart come sistema init

In AL2 `upstart` è stato sostituito da `systemd` as the `init` system. AL2023 utilizza anche `systemd` come `init` sistema, adottando ulteriormente nuove caratteristiche e funzionalità di `systemd`.

Python 2.6 e 2.7 sono stati sostituiti con Python 3

Sebbene Python 2.6 fosse AL1 contrassegnato come EOL nella versione 2018.03, i pacchetti erano ancora disponibili nei repository per l'installazione. AL2 fornito con Python 2.7 come prima versione di Python supportata AL2 e 023 completa la transizione a Python 3. Nessuna versione di Python 2.x è inclusa nei repository 023. AL2

Per ulteriori informazioni su Python su Amazon Linux, consulta [Pythonnel AL2023](#).

OpenJDK 8 come JDK più vecchio

AL2023 viene fornito con [Amazon Corretto](#) come Java Development Kit (JDK) predefinito (e unico). Tutti Java i pacchetti basati in AL2 023 sono compilati con Amazon Corretto 17.

Nel AL1, `java-1.6.0-openjdk` () è diventato EOL con la prima versione 2018.03 e `java-1.7.0-openjdk` () è diventato EOL a metà 2020, sebbene entrambe le versioni fossero disponibili nei repository. `java-1.7.0-openjdk` AL1 La prima versione di OpenJDK disponibile nel AL2 023 è OpenJDK 8, fornita da Amazon Corretto 8.

AL2023 modifiche al kernel rispetto ad Amazon Linux 1 () AL1

Applicazione di patch live del kernel

Entrambi AL2023 e AL2 aggiungono il supporto per la funzionalità di live-patching del kernel. Ciò consente di correggere vulnerabilità di sicurezza critiche e importanti nel kernel Linux senza riavvio o tempi di inattività. Per ulteriori informazioni, consulta [Kernel Live Patching attivo AL2023](#).

Supporto del file system del kernel

Sono state apportate diverse modifiche ai file system che il kernel AL1 supporterà il montaggio, oltre a cambiamenti negli schemi di partizionamento che il kernel analizzerà.

Opzione CONFIG	AL1/4.14/ x86_64	AL2023/6. 1/ aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/ aarch64	AL2023/6. 12/ x86_64	AL2023/6. 18/ aarch64	AL2023/6. 18/ x86_64
<u>CONFIG_AISFS</u>	m	n	n	n	n	n	n
<u>CONFIG_AISRPC</u>	m	n	n	n	n	n	n
<u>CONFIG_BSD_DISKLABEL</u>	y	n	n	n	n	n	n
<u>CONFIG_CIFS</u>	m	n	n	n	n	n	n
<u>CONFIG_CIFS_BLOCKDEV</u>	N/D	N/D	N/D	N/D	N/D	N/D	N/D
<u>CONFIG_CIFS_CLONE</u>	N/D	n	n	n	n	n	n
<u>CONFIG_CIFS_ERA</u>	n	n	n	n	n	n	n
<u>CONFIG_CIFS_INTEGRITY</u>	m	m	m	m	m	m	m
<u>CONFIG_CIFS_LOG_WRITES</u>	n	m	m	m	m	m	m
<u>CONFIG_CIFS_SWITCH</u>	n	n	n	n	n	n	n

Opzione CONFIG	AL1/4.14/ x86_64	AL2023/6. 1/ aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/ aarch64	AL2023/6. 12/ x86_64	AL2023/6. 18/ aarch64	AL2023/6. 18/ x86_64
<u>CONFIG_DEBUG_KERNEL</u>	n	m	m	m	m	m	m
<u>CONFIG_CRYPTO_DEV_CRYPTOFS</u>	m	n	n	n	n	n	n
<u>CONFIG_CRYPTO_FATFS</u>	N/D	m	m	m	m	m	m
<u>CONFIG_CRYPTO_T2FS</u>	m	n	n	n	n	n	n
<u>CONFIG_CRYPTO_T3FS</u>	m	n	n	n	n	N/D	N/D
<u>CONFIG_GS2FS</u>	n	n	n	n	n	n	n
<u>CONFIG_HISPLUSFS</u>	m	n	n	n	n	n	n
<u>CONFIG_HISFS</u>	m	n	n	n	n	n	n
<u>CONFIG_JFSFS</u>	n	n	n	n	n	n	n
<u>CONFIG_LIMPARTITION</u>	y	n	n	n	n	n	n
<u>CONFIG_MPCPARTITION</u>	y	n	n	n	n	n	n

Opzione CONFIG	AL1/4.14/ x86_64	AL2023/6. 1/ aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/ aarch64	AL2023/6. 12/ x86_64	AL2023/6. 18/ aarch64	AL2023/6. 18/ x86_64
CONFIG_NAMES_V2	m	n	n	n	n	n	n
CONFIG_NAMES_FS_FS	m	n	n	n	n	n	n
CONFIG_NAMES_MFS_FS	m	n	n	n	n	n	n
CONFIG_SECURITY_LARIS_X86_PARTITION	y	n	n	n	n	n	n
CONFIG_SECURITY_UASHFS_ZSTD	y	y	y	y	y	y	y
CONFIG_SECURITY_PARTITION	y	n	n	n	n	n	n

Modifiche alla configurazione del kernel incentrate sulla sicurezza

Opzione CONFIG	AL1/4.14/ x86_64	AL2023/6. 1/ aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/ aarch64	AL2023/6. 12/ x86_64	AL2023/6. 18/ aarch64	AL2023/6. 18/ x86_64
CONFIG_BUG_ON_DATA_CORRUPTION	y	y	y	y	y	y	y

Opzione CONFIG	AL1/4.14/ x86_64	AL2023/6. 1/ aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/ aarch64	AL2023/6. 12/ x86_64	AL2023/6. 18/ aarch64	AL2023/6. 18/ x86_64
CONFIG_DEBUG_PAGE_REF	4096	65536	65536	65536	65536	65536	65536
CONFIG_DEBUG_VIRTUAL	y	n	n	n	n	n	n
CONFIG_DEBUG_VPORT	y	n	n	n	n	n	n
CONFIG_DEBUG_USER_SOURCE	y	y	y	y	y	y	y
CONFIG_DEBUG_USER_COPY_FALLBACK	N/D	N/D	N/D	N/D	N/D	N/D	N/D
CONFIG_DEBUG_INIT_ON_ALLOC_DEFAULT_ON	N/D	n	n	n	n	n	n
CONFIG_DEBUG_INIT_ON_FREE_DEFAULT_ON	N/D	n	n	n	n	n	n
CONFIG_DEBUG_IOCTL_DEFAULT	N/D	n	n	n	n	n	n

Opzione CONFIG	AL1/4.14/ x86_64	AL2023/6. 1/ aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/ aarch64	AL2023/6. 12/ x86_64	AL2023/6. 18/ aarch64	AL2023/6. 18/ x86_64
CONFIG_L ISC_AUTO OAD	y	n	n	n	n	n	n
CONFIG_SC HED_CORE	N/D	N/D	y	N/D	y	N/D	y
CONFIG_SC HED_STACK _END_CHEC K	y	y	y	y	y	y	y
CONFIG_SI CURITY_DI ESG_RESTI ICT	n	y	y	y	y	y	y
CONFIG_SI CURITY_SI LINUX_DIS ABLE	y	n	n	N/D	N/D	N/D	N/D
CONFIG_SI UFFLE_PAC E_ALLOCA OR	N/D	y	y	y	y	y	y
CONFIG_SI AB_FREEL ST_HARDE ED	y	y	y	y	y	y	y

Opzione CONFIG	AL1/4.14/ x86_64	AL2023/6. 1/ aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/ aarch64	AL2023/6. 12/ x86_64	AL2023/6. 18/ aarch64	AL2023/6. 18/ x86_64
CONFIG_SIB AB_FREEL ST_RANDOM	n	y	y	y	y	y	y

Altre modifiche apportate alla configurazione del kernel

Opzione CONFIG	AL1/4.14/ x86_64	AL2023/6. 1/ aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/ aarch64	AL2023/6. 12/ x86_64	AL2023/6. 18/ aarch64	AL2023/6. 18/ x86_64
CONFIG_HZ	250	100	100	100	100	100	100
CONFIG_NR _CPUS	8192	4096	8192	4096	8192	4096	8192
CONFIG_P NIC_ON_O PS	n	y	y	y	y	y	y
CONFIG_P NIC_ON_O PS_VALUE	0	1	1	1	1	N/D	N/D
CONFIG_PI P	m	m	m	m	m	m	m
CONFIG_SI IP	m	n	n	n	n	n	n
CONFIG_XI N_PV	y	N/D	n	N/D	n	N/D	n

Confronto dei pacchetti installati su Amazon Linux 1 (AL1) e Amazon Linux 2023 AMIs

Un confronto tra lo standard RPMs attuale AL1 e AL2 lo standard 023. AMIs

Pacchetto	AL1 AMI	AL2023 AMI
acl	2,2,49	2.3.1
acpid	2.0,19	2.0,32
alsa-lib	1,0,22	
alternatives		1.15
amazon-chrony-config		4.3
amazon-ec2-net-utils		2.5.1
amazon-linux-repo-s3		2023,620241031
amazon-linux-sb-keys		2023,1
amazon-rpm-config		228
amazon-ssm-agent	3,22222,0	3,3,987,0
amd-ucode-firmware		20210208
at	3,1,10	3,1,23
attr	2,4,46	2.5.1
audit	2,6,5	30.6
audit-libs	2,6,5	30.6
authconfig	6.2.8	
aws-amitools-ec2	1,5,13	

Pacchetto	AL1 AMI	AL2023 AMI
aws-cfn-bootstrap	1.4	2.0
aws-cli	1,118,107	
awscli-2		2,15,30
basesystem	10,0	11
bash	4,2,46	5,2,15
bash-completion		2.11
bc	1,006,95	1,07,1
bind-libs	9,8,2	9,18,28
bind-license		9,18,28
bind-utils	9,82	9,18,28
binutils	2,27	2,39
boost-filesystem		1,75,0
boost-system		1,75,0
boost-thread		1,75,0
bzip2	1.0.6	1.0.8
bzip2-libs	1.0.6	1.0.8
ca-certificates	2023,2,62	2023,2,68
c-ares		1.19.1
checkpolicy	2.1.10	3.4
chkconfig	1,349,3	1.15

Pacchetto	AL1 AMI	AL2023 AMI
chrony		4.3
cloud-disk-utils	0,27	
cloud-init	0,7,6	222,2
cloud-init-cfg-ec2		222,2
cloud-utils-growpart		0,31
copy-jdk-configs	3.3	
coreutils	8,22	8,32
coreutils-common		8,32
cpio	(2.10)	2,13
cracklib	2,8,16	2,9,6
cracklib-dicts	2,8,16	2,9,6
cronie	1.4.4	
cronie-anacron	1.4.4	
crontabs	1.10	1.11
crypto-policies		20220428
crypto-policies-scripts		20220428
cryptsetup	1,6,7	2.6.1
cryptsetup-libs	1.6.7	2.6.1
curl	7,61,1	
curl-minimal		8,5,0

Pacchetto	AL1 AMI	AL2023 AMI
cyrus-sasl	2,1,23	
cyrus-sasl-lib	2,1,23	2,1,27
cyrus-sasl-plain	2,1,23	2,1,27
dash	0,5,5,1	
db4	4,7,25	
db4-utils	4,7,25	
dbus	1,6,12	1,12,28
dbus-broker		32
dbus-common		1,12,28
dbus-libs	1,6,12	1,12,28
dejavu-fonts-common	2,33	
dejavu-sans-fonts	2,33	
dejavu-serif-fonts	2,33	
device-mapper	1,02,135	1,02,185
device-mapper-event	1,02,135	
device-mapper-event-libs	1,02,135	
device-mapper-libs	1,02,135	1,02,185
device-mapper-persistent-data	0,63	
dhclient	4.1.1	

Pacchetto	AL1 AMI	AL2023 AMI
dhcp-common	4.1.1	
diffutils	3.3	3.8
dmraid	1.0.0.rc16	
dmraid-events	1.0.0.rc16	
dnf		4.14.0
dnf-data		4.14.0
dnf-plugin-release-notification		1.2
dnf-plugins-core		4.3.0
dnf-plugin-support-info		1.2
dnf-utils		4.3.0
dosfstools		4.2
dracut	004	055
dracut-config-ec2		3.0
dracut-config-generic		055
dracut-modules-growroot	0.20	
dump	0.4	
dwz		0,14
dyninst		102,1

Pacchetto	AL1 AMI	AL2023 AMI
e2fsprogs	1,43,5	1,46,5
e2fsprogs-libs	1,43,5	1,46,5
ec2-hibinit-agent	1.0.0	1.0.8
ec2-instance-connect		1.1
ec2-instance-connect-selinux		1.1
ec2-net-utils	0.7	
ec2-utils	0.7	2.2.0
ed	1.1	1.14.2
efi-filesystem		5
efi-srpm-macros		5
efivar		38
efivar-libs		38
elfutils-debuginfod-client		0.188
elfutils-default-yama-scope		0.188
elfutils-libelf	0,168	0.188
elfutils-libs		0.188
epel-release	6	
ethtool	3,15	5,15

Pacchetto	AL1 AMI	AL2023 AMI
expat	2.1.0	2.5.0
file	5,37	5,39
file-libs	5,37	5,39
filesystem	2,4,30	3,14
findutils	44,2	4.8.0
fipscheck	1.3.1	
fipscheck-lib	1.3.1	
fontconfig	2.8.0	
fontpackages-files system	1,41	
fonts-srpm-macros		2.0.5
freetype	2,3,11	
fstrm		0.6.1
fuse-libs	2,9,4	2,9,9
gawk	3.1.7	5.1.0
gdbm	1.8.0	
gdbm-libs		1,19
gdisk	0,8,10	1.0.8
generic-logos	17,0	
get_reference_source	1.2	
gettext		0,21

Pacchetto	AL1 AMI	AL2023 AMI
gettext-libs		0,21
ghc-srpm-macros		1.5.0
giflib	4.1.6	
glib2	2,36,3	2,74,7
glibc	2,17	2,34
glibc-all-langpacks		2,34
glibc-common	2,17	2,34
glibc-gconv-extra		2,34
glibc-locale-source		2,34
gmp	6.0.0	62,1
gnupg2	2.0,28	
gnupg2-minimal		2.3.7
gnutls		3.8.0
go-srpm-macros		3.2.0
gpgme	1.4.3	1.15.1
gpm-libs	1,20,6	1,20.7
grep	2,20	3.8
groff	1,22.2	
groff-base	1,22.2	1,22.4
grub	0,97	

Pacchetto	AL1 AMI	AL2023 AMI
grub2-common		2,06
grub2-efi-x64-ec2		2,06
grub2-pc-modules		2,06
grub2-tools		2,06
grub2-tools-minimal		2,06
grubby	7,0,15	8,40
gssproxy		0.8.4
gzip	1.5	1.12
hesiod	3.1.0	
hibagent	1.0.0	
hmaccalc	0,9,12	
hostname		3,23
hunspell		1.7.0
hunspell-en		0,20140811,1
hunspell-en-GB		0,20140811,1
hunspell-en-US		0,20140811,1
hunspell-filesystem		1.7.0
hwdata	0,233	0,384
info	5.1	6.7
inih		49

Pacchetto	AL1 AMI	AL2023 AMI
initscripts	9,03,58	10,09
iproute	4.4.0	6.10.0
iptables	1,4,21	
iputils	20121221	20210202
irqbalance	1.5.0	1.9.0
jansson		2.14
java-1.7.0-openjdk	1,7,0,321	
javapackages-tools	0.9.1	
jemalloc		5.2.1
jitterentropy		3.4.1
jpackage-utils	1,7,5	
jq		1.7.1
json-c		0,14
kbd	1.15	2.4.0
kbd-misc	1.15	2.4.0
kernel	4,14,336	6,1112
kernel-libbpf		6,1112
kernel-livepatch-r epo-s3		2023,620241031
kernel-srpm-macros		1.0
kernel-tools	4,14,336	6,1112

Pacchetto	AL1 AMI	AL2023 AMI
keyutils	1.5.8	1.6.3
keyutils-libs	1.5.8	1.6.3
kmod	14	29
kmod-libs	14	29
kpartx	0,49	
kpatch-runtime		0,9,7
krb5-libs	1.15.1	1,21,3
lcms2	2.6	
less	436	608
libacl	2,2,49	2.3.1
libaio	0,3109	0,3111
libarchive		3,7,4
libargon2		20171227
libassuan	2.0.3	2,5,5
libattr	2,4,46	2.5.1
libbasicobjects		0,11
libblkid	2,23,2	2,37,4
libcap	2,16	2,48
libcap54	2,54	
libcap-ng	0,7,5	0.8.2

Pacchetto	AL1 AMI	AL2023 AMI
libcbor		0.7.0
libcgroup	0,40. rc1	
libcollection		0.7.0
libcom_err	1,43,5	1,46,5
libcomps		01,20
libconfig		17.2
libcurl	7,61,1	
libcurl-minimal		8,5,0
libdb		5,3,28
libdhash		0,50
libdnf		0,69,0
libeconf		0,40
libedit	2.11	3.1
libev		4,33
libevent	2.0,21	2,1,12
libfdisk		2,37,4
libffi	3,0,13	34.4
libfido2		1.10.0
libfontenc	1.0.5	
libgcc		11.4.1

Pacchetto	AL1 AMI	AL2023 AMI
libgcc72	7.2.1	
libgcrypt	1.5.3	1.10.2
libgomp		11.4.1
libgpg-error	1.11	1,42
libgssglue	0.1	
libibverbs		48,0
libICE	1.0.6	
libicu	50,2	
libidn	1,18	
libidn2	2.3.0	2.3.2
libini_config		1.3.1
libjpeg-turbo	1,2,90	
libkcap		1.4.0
libkcap-hmacalc		1.4.0
libldb		26.2
libmaxinddb		1.5.2
libmetalink		0,13
libmnl	1.0.3	1.0.4
libmodulemd		2.13.0
libmount	2,23,2	2,37,4

Pacchetto	AL1 AMI	AL2023 AMI
libnetfilter_contrack	1.0.4	
libnfnetworking	1.0.1	
libnfsidmap	0.25	2,5,4
libnghttp2	1,33,0	1,59,0
libnih	1.0.1	
libnl	1.1.4	
libnl3		3.5.0
libpath_utils		0,2,1
libpcap		1.10.1
libpipeline	1.2.3	1.5.3
libpkgconf		1.8.0
libpng	1,2,49	
libpsl	0.6.2	0,21,1
libpwquality	1.2.3	1.4.4
libref_array		0,1,5
librepo		1,14,5
libreport-filesystem		2,15,2
libseccomp		2.5.3
libselinux	2.1.10	3.4
libselinux-utils	2.1.10	3.4

Pacchetto	AL1 AMI	AL2023 AMI
libsemanage	2.1.6	3.4
libsepol	2.1.7	3.4
libsigsegv		2,13
libSM	1.2.1	
libsmartcols	2,23,2	2,37,4
libsolv		0,7,22
libss	1,43,5	1,46,5
libssh2	1.4.2	
libsss_certmap		2,9,4
libsss_idmap		2,9,4
libsss_nss_idmap		2,9,4
libsss_sudo		2,9,4
libstdc++		11.4.1
libstdc++72	7.2.1	
libstoragegmt		1.9.4
libsysfs	2.1.0	
libtalloc		2.3.4
libtasn1	2.3	4,19,0
libtdb		1.4.7
libtevent		0.13.0

Pacchetto	AL1 AMI	AL2023 AMI
libtextstyle		0,21
libtirpc	0,24	1.3.3
libudev	173	
libunistring	0,9,3	0,9,10
libuser	0,60	0,63
libutempter	1.1.5	1.2.1
libuuid	2,23,2	2,37,4
libuv		1.47.0
libverto	0,2,5	0,32
libverto-libev		0,32
libX11	1.6.0	
libX11-common	1.6.0	
libXau	1.0.6	
libxcb	1.11	
libXcomposite	0,4,3	
libxcrypt		4,4,33
libXext	1.3.2	
libXfont	1.4.5	
libXi	1,7,2	
libxml2	29.1	210.4

Pacchetto	AL1 AMI	AL2023 AMI
libxml2-python27	29.1	
libXrender	0,9,8	
libxslt	1,1,28	
libXtst	1.2.2	
libyaml	0,16	02,5
libzstd		1,5,5
linux-firmware-whe nce		20210208
lm_sensors-libs		3.6.0
lmdb-libs		0,9,29
log4j-cve-2021-442 28-hotpatch	1.3	
logrotate	3,78	3,20,1
lsyf	4,82	4,94,0
lua	5.1.4	
lua-libs		5.4.4
lua-srpm-macros		1
lvm2	2,02,166	
lvm2-libs	2,02,166	
lz4-libs		1.9.4
mailcap	2,1,31	

Pacchetto	AL1 AMI	AL2023 AMI
make	3,82	
man-db	26.3	29.3
man-pages	4,10	5,10
mdadm	3.2.6	
microcode_ctl	2.1	2.1
mingetty	1,08	
mpfr		4.1.0
nano	2.5.3	5.8
nc	1,84	
ncurses	5.7	6.2
ncurses-base	5.7	6.2
ncurses-libs	5.7	6.2
nettle		3.8
net-tools	1,60	2.0
newt	0,52,11	0,52,21
newt-python27	0,52,11	
nfs-utils	1.3.0	2,5,4
npth		1.6
nspr	4,25,0	4,35,0
nss	3,53,1	3,90,0

Pacchetto	AL1 AMI	AL2023 AMI
nss-pem	1.0.3	
nss-softokn	3,53,1	3,90,0
nss-softokn-freebl	3,53,1	3,90,0
nss-sysinit	3,53,1	3,90,0
nss-tools	3,53,1	
nss-util	3,53,1	3,90,0
ntp	4.2.8 p15	
ntpdate	4.2.8p15	
ntsysv	1,349,3	1.15
numactl	2.0.7	
numactl-libs		2.0,14
ocaml-srpm-macros		6
oniguruma		6,9,7,1
openblas-srpm-macros		2
openldap	2,4,40	2,4,57
openssh	7,4p 1	8,7p1
openssh-clients	7,4p1	8,7p1
openssh-server	7,4p1	8,7p1
openssl	1,0,2k	3,0,8
openssl-libs		30,8

Pacchetto	AL1 AMI	AL2023 AMI
openssl-pkcs11		0,4,12
os-prober		1,77
p11-kit	0,18,5	0,24,1
p11-kit-trust	0,18,5	0,24,1
package-notes-srpm-macros		0.4
pam	1.1.8	1.5.1
pam_ccreds	10	
pam_krb5	2,3,11	
pam_passwdqc	1.0.5	
parted	2.1	3.4
passwd	0,79	0,80
pciutils	3,1,10	3.7.0
pciutils-libs	3,1,10	3.7.0
pcre	8,21	
pcre2		10,40
pcre2-syntax		10,40
perl	5,16,3	
perl-Carp	1,26	1,50
perl-Class-Struct		0,66
perl-constant	1,27	1,33

Pacchetto	AL1 AMI	AL2023 AMI
perl-Digest	1,17	
perl-Digest-HMAC	1,03	
perl-Digest-MD5	2,52	
perl-Digest-SHA	5,85	
perl-DynaLoader		1,47
perl-Encode	2,51	3,15
perl-Errno		1,30
perl-Exporter	5,68	5,74
perl-Fcntl		1.13
perl-File-Basename		2,85
perl-File-Path	2,09	2,18
perl-File-stat		1,09
perl-File-Temp	0,23,01	0,231,100
perl-Filter	1,49	
perl-Getopt-Long	2,40	2,52
perl-Getopt-Std		1.12
perl-HTTP-Tiny	0,033	0,078
perl-if		0,60,800
perl-interpreter		5,32,1
perl-IO		1,43

Pacchetto	AL1 AMI	AL2023 AMI
perl-IPC-Open3		1,21
perl-libs	5,16,3	5,32,1
perl-macros	5,16,3	
perl-MIME-Base64		3,16
perl-mro		1,23
perl-overload		1,31
perl-overloading		0,02
perl-parent	0,225	0,238
perl-PathTools	3,40	3,78
perl-Pod-Escapes	1.04	1,07
perl-podlators	2.5.1	4,14
perl-Pod-Perldoc	3,20	3,28,01
perl-Pod-Simple	3,28	3,42
perl-Pod-Usage	1,63	2,01
perl-POSIX		1,94
perl-Scalar-List-Utils	1,27	1,56
perl-SelectSaver		1.02
perl-Socket	2,010	2,032
perl-srpm-macros		1
perl-Storable	2,45	3,21

Pacchetto	AL1 AMI	AL2023 AMI
perl-subst		1,03
perl-Symbol		1,08
perl-Term-ANSIColor		5,01
perl-Term-Cap		1,17
perl-Text-ParseWords	3,29	3,30
perl-Text-Tabs+Wrap		2021,0726
perl-threads	1,87	
perl-threads-shared	1,43	
perl-Time-HiRes	1,9725	
perl-Time-Local	1,2300	1.300
perl-vars		1,05
pinentry	0,7,6	
pkgconf		1.8.0
pkgconfig	0,27,1	
pkgconf-m4		1.8.0
pkgconf-pkg-config		1.8.0
pm-utils	1.4.1	
policycoreutils	2,1,12	3.4
policycoreutils-python-utils		3.4
popt	1.13	1,18

Pacchetto	AL1 AMI	AL2023 AMI
procmail	3,22	
procps	32,8	
procps-ng		3,3,17
protobuf-c		1.4.1
psacct	6,32	6.6.4
psmisc	22,20	23,4
pth	2.0.7	
publicsuffix-list-dafsa		20240212
python27	2,7,18	
python27-babel	0.9.4	
python27-backports	1.0	
python27-backports-ssl_match_hostname	3,40,2	
python27-boto	2,48,0	
python27-botocore	1,17,31	
python27-chardet	2.0.1	
python27-colorama	0,4,1	
python27-configobj	4,7,2	
python27-crypto	2.6.1	
python27-daemon	1.5.2	

Pacchetto	AL1 AMI	AL2023 AMI
python27-dateutil	2.1	
python27-devel	2,7,18	
python27-docutils	0,11	
python27-ecdsa	0,11	
python27-futures	3.0.3	
python27-imaging	1.1.6	
python27-iniparse	0,31	
python27-jinja2	27.2	
python27-jmespath	0.9.2	
python27-jsonpatch	1.2	
python27-jsonpointer	1.0	
python27-kitchen	1.1.1	
python27-libs	2,7,18	
python27-lockfile	0.8	
python27-markupsafe	0,11	
python27-paramiko	1.15.1	
python27-pip	9,0,3	
python27-ply	3.4	
python27-pyasn1	0,17	
python27-pycurl	7,19,0	

Pacchetto	AL1 AMI	AL2023 AMI
python27-pygpme	0.3	
python27-pyliblzma	0,5,3	
python27-pystache	0,5,3	
python27-pyxattr	0,50	
python27-PyYAML	3,10	
python27-requests	1.2.3	
python27-rsa	3.4.1	
python27-setuptools	36,27	
python27-simplejson	3.6.5	
python27-six	1.8.0	
python27-urlgrabber	3,10	
python27-urllib3	1,24,3	
python27-virtualenv	15,10	
python3		3,9,16
python3-attrs		20,3,0
python3-audit		30.6
python3-awscli		0,19,19
python3-babel		2,9,1
python3-cffi		1,14,5
python3-chardet		4.0.0

Pacchetto	AL1 AMI	AL2023 AMI
python3-colorama		04.4
python3-configobj		50.6
python3-cryptography		36,0
python3-daemon		2.3.0
python3-dateutil		28.1
python3-dbus		1,2,18
python3-distro		1.5.0
python3-dnf		4.14.0
python3-dnf-plugins-core		4.3.0
python3-docutils		0,16
python3-gpg		1.15.1
python3-hawkey		0,69,0
python3-idna		(2.10)
python3-jinja2		211,3
python3-jmespath		0.10.0
python3-jsonpatch		1,21
python3-jsonpointer		2.0
python3-jsonschemata		3.2.0
python3-libcomps		01,20
python3-libdnf		0,69,0

Pacchetto	AL1 AMI	AL2023 AMI
python3-libs		3,9,16
python3-libselinux		3.4
python3-libsemanage		3.4
python3-libstorage mgmt		1.9.4
python3-lockfile		0.12.2
python3-markupsafe		1.1.1
python3-netifaces		0,10,6
python3-oauthlib		3.0.2
python3-pip-wheel		21,31
python3-ply		3,11
python3-policycore utils		3.4
python3-prettytable		0.7.2
python3-prompt-too lkit		3,0,24
python3-pycparser		2,20
python3-pyrsistent		0,17,3
python3-pyserial		3.4
python3-pysocks		1.7.1
python3-pytz		2022,7,1

Pacchetto	AL1 AMI	AL2023 AMI
python3-pyyaml		5.4.1
python3-requests		2,25,1
python3-rpm		4,161,3
python3-ruamel-yaml		0,16,6
python3-ruamel-yaml-clib		0,12
python3-setools		4.4.1
python3-setuptools		59,6,0
python3-setuptools-wheel		59,6,0
python3-six		1.15.0
python3-systemd		235
python3-urllib3		1,25,10
python3-wcwidth		0,2,5
python-chevront		0.13.1
python-srpm-macros		3.9
quota	4,00	4,06
quota-nls	4,00	4,06
readline	6.2	8.1
rmt	0.4	
rng-tools	5	6,14

Pacchetto	AL1 AMI	AL2023 AMI
rootfiles	8.1	8.1
rpcbind	0,2,0	1.2.6
rpm	4.11.3	4,161,3
rpm-build-libs	411,3	4,161,3
rpm-libs	411,3	4,161,3
rpm-plugin-selinux		4,161,3
rpm-plugin-systemd-inhibit		4,161,3
rpm-python27	411,3	
rpm-sign-libs		4,161,3
rsync	30.6	3.2.6
rsyslog	5,8,10	
ruby	2.0	
ruby20	2,0,0,648	
ruby20-irb	2,0,0,648	
ruby20-libs	2,0,0,648	
rubygem20-bigdecimal	1.2.0	
rubygem20-json	1.8.3	
rubygem20-psych	2.0.0	
rubygem20-rdoc	42,2	
rubygems20	2,014,1	

Pacchetto	AL1 AMI	AL2023 AMI
rust-srpm-macros		21
sbsigntools		0.9.4
screen	40,3	4.8.0
sed	4.2.1	4.8
selinux-policy		381,45
selinux-policy-targeted		381,45
sendmail	8,14,4	
setserial	2,17	
setup	2,8,14	2,13,7
sgpio	1,2,0,10	
shadow-utils	4,14.2	4.9
shared-mime-info	1.1	
slang	2.2.1	2.3.2
sqlite	3,7,17	
sqlite-libs		3,4,0
sssd-client		2,9,4
sssd-common		2,9,4
sssd-kcm		2,9,4
sssd-nfs-idmap		2,9,4
strace		6.8

Pacchetto	AL1 AMI	AL2023 AMI
sudo	1,8,23	1,9,15
sysctl-defaults	1.0	1.0
sysfsutils	2.1.0	
sysstat		12,5,6
systemd		252,23
systemd-libs		252,23
systemd-networkd		252,23
systemd-pam		252,23
systemd-resolved		252,23
systemd-udev		252,23
system-release	2018,03	2023,620241031
systemtap-runtime		4,8
sysvinit	2,87	
tar	1,26	1,34
tbb		2020,3
tcp_wrappers	7.6	
tcp_wrappers-libs	7.6	
tcpdump		4,99,1
tcsh		6,24,07
time	1,7	1.9

Pacchetto	AL1 AMI	AL2023 AMI
tmpwatch	2,9,16	
traceroute	2.0,14	2.1.3
ttmkfdir	3,09	
tzdata	2023 c	2024a
tzdata-java	2023c	
udev	173	
unzip	6.0	6.0
update-motd	1.0.1	2.2
upstart	0,6,5	
userspace-rcu		0.12.1
ustr	1.0.4	
util-linux	2,23,2	2,37,4
util-linux-core		2,37,4
vim-common	9,0,2120	9,0,2153
vim-data	9,0,2120	9,0,2153
vim-enhanced	9,0,2120	9,0,2153
vim-filesystem	9,0,2120	9,0,2153
vim-minimal	9,0,2120	9,0,2153
wget	1,18	1,21,3
which	2,19	2,21

Pacchetto	AL1 AMI	AL2023 AMI
words	3.0	3.0
xfsdump		3,1,11
xfspgrog		5,18,0
xorg-x11-fonts-Type1	7.2	
xorg-x11-font-utils	7.2	
xxd	9,0,2120	9,0,2153
xxhash-libs		0.8.0
xz	5.2.2	5.2.5
xz-libs	5.2.2	5.2.5
yum	3.4.3	4.14.0
yum-metadata-parser	1.1.4	
yum-plugin-priorities	1,1,31	
yum-plugin-upgrade-helper	1,1,31	
yum-utils	1,1,31	
zip	3.0	3.0
zlib	1.2.8	1,2,11
zram-generator		1.1.2
zram-generator-defaults		1.1.2

Pacchetto	AL1 AMI	AL2023 AMI
zstd		1,5,5

Confronto dei pacchetti installati su Amazon Linux 1 (AL1) e Amazon Linux 2023 Minimal AMIs

Un confronto tra il RPMs presente AL1 e AL2 lo 023 Minimal. AMIs

Pacchetto	AL1 Minimo	AL2023 Minimo
acpid	2.0.19	
alternatives		1.15
amazon-chrony-config		4.3
amazon-ec2-net-utils		2.5.1
amazon-linux-repo-s3		2023,620241031
amazon-linux-sb-keys		2023,1
amd-ucode-firmware		20210208
audit	2,6,5	30.6
audit-libs	2,6,5	30.6
authconfig	6.2.8	
awscli-2		2,15,30
basesystem	10,0	11
bash	4,2,46	5,2,15
binutils	2,27	

Pacchetto	AL1 Minimo	AL2023 Minimo
bzip2	1.0.6	
bzip2-libs	1.0.6	1.0.8
ca-certificates	2023,2,62	2023,2,68
checkpolicy	2.1.10	3.4
chkconfig	1,349,3	
chrony		4.3
cloud-disk-utils	0,27	
cloud-init	0,7,6	222,2
cloud-init-cfg-ec2		222,2
cloud-utils-growpart		0,31
coreutils	8,22	8,32
coreutils-common		8,32
cpio	(2.10)	2,13
cracklib	2,8,16	2,9,6
cracklib-dicts	2,8,16	2,9,6
cronie	1.4.4	
cronie-anacron	1.4.4	
crontabs	1.10	
crypto-policies		20220428
cryptsetup-libs		2.6.1

Pacchetto	AL1 Minimo	AL2023 Minimo
curl	7,61,1	
curl-minimal		8,5,0
cyrus-sasl	2,1,23	
cyrus-sasl-lib	2,1,23	2,1,27
dash	0,5,5,1	
db4	4,7,25	
db4-utils	4,7,25	
dbus		1,12,28
dbus-broker		32
dbus-common		1,12,28
dbus-libs	1,6,12	1,12,28
device-mapper		1,02,185
device-mapper-libs		1,02,185
dhclient	4.1.1	
dhcp-common	4.1.1	
diffutils	3.3	3.8
dnf		4.14.0
dnf-data		4.14.0
dnf-plugin-release-notification		1.2
dnf-plugins-core		4.3.0

Pacchetto	AL1 Minimo	AL2023 Minimo
dnf-plugin-support-info		1.2
dracut	004	055
dracut-config-ec2		3.0
dracut-config-generic		055
dracut-modules-growroot	0.20	
e2fsprogs	1,43,5	1,46,5
e2fsprogs-libs	1,43,5	1,46,5
ec2-utils	0.7	2.2.0
ed	1.1	
efi-filesystem		5
efivar		38
efivar-libs		38
elfutils-default-yama-scope		0.188
elfutils-libelf	0,168	0.188
elfutils-libs		0.188
ethtool	3,15	
expat	2.1.0	2.5.0
file	5,37	5,39

Pacchetto	AL1 Minimo	AL2023 Minimo
file-libs	5,37	5,39
filesystem	2,4,30	3,14
findutils	44,2	4.8.0
fipscheck	1.3.1	
fipscheck-lib	1.3.1	
fuse-libs	2,9,4	2,9,9
gawk	3.1.7	5.1.0
gdbm	1.8.0	
gdbm-libs		1,19
gdisk	0,8,10	1.0.8
generic-logos	17,0	
get_reference_source	1.2	
gettext		0,21
gettext-libs		0,21
glib2	2,36,3	2,74,7
glibc	2,17	2,34
glibc-all-langpacks		2,34
glibc-common	2,17	2,34
glibc-locale-source		2,34
gmp	6.0.0	62,1

Pacchetto	AL1 Minimo	AL2023 Minimo
gnupg2	2.0,28	
gnupg2-minimal		2.3.7
gnutls		3.8.0
gpgme	1.4.3	1.15.1
grep	2,20	3.8
groff	1,22.2	
groff-base	1,22.2	1,22.4
grub	0,97	
grub2-common		2,06
grub2-efi-x64-ec2		2,06
grub2-pc-modules		2,06
grub2-tools		2,06
grub2-tools-minimal		2,06
grubby	7,0,15	8,40
gzip	1.5	1.12
hesiod	3.1.0	
hmacalc	0,9,12	
hostname		3,23
hwdata	0,233	0,384
info	5.1	

Pacchetto	AL1 Minimo	AL2023 Minimo
inih		49
initscripts	9,03,58	10,09
iproute	4.4.0	6.10.0
iptables	1,4,21	
iputils	20121221	20210202
irqbalance		1.9.0
jansson		2.14
jitterentropy		3.4.1
jq		1.7.1
json-c		0,14
kbd	1.15	2.4.0
kbd-misc	1.15	2.4.0
kernel	4,148336	6,1112
kernel-libbpf		6,1112
kernel-livepatch-r epo-s3		2023,620241031
keyutils-libs	1,58	1.6.3
kmod	14	29
kmod-libs	14	29
krb5-libs	1.15.1	1,21,3
less	436	608

Pacchetto	AL1 Minimo	AL2023 Minimo
libacl	2,2,49	2.3.1
libarchive		3,74
libargon2		20171227
libassuan	2.0.3	2,5,5
libattr	2,4,46	2.5.1
libblkid	2,23,2	2,37,4
libcap	2,16	2,48
libcap54	2,54	
libcap-ng	0,7,5	0.8.2
libcbor		0.7.0
libcgroup	0,40. rc1	
libcom_err	1,43,5	1,46,5
libcomps		01,20
libcurl	7,61,1	
libcurl-minimal		8,5,0
libdb		5,3,28
libdnf		0,69,0
libeconf		0,40
libedit	2.11	3.1
libfdisk		2,37,4

Pacchetto	AL1 Minimo	AL2023 Minimo
libffi	3,0,13	34.4
libfido2		1.10.0
libgcc		114,1
libgcc72	7.2.1	
libgcrypt	1.5.3	1.10.2
libgomp		11.4.1
libgpg-error	1.11	1,42
libicu	50,2	
libidn	1,18	
libidn2	2.3.0	2.3.2
libkcapi		1.4.0
libkcapi-hmaccalc		1.4.0
libmnl	1.0.3	1.0.4
libmodulemd		2.13.0
libmount	2,23,2	2,37,4
libnetfilter_contrack	1.0.4	
libnfnetlink	1.0.1	
libnhttp2	1,33,0	1,59,0
libnih	1.0.1	
libpipeline		1.5.3

Pacchetto	AL1 Minimo	AL2023 Minimo
libpsl	0.6.2	0,21,1
libpwquality	1.2.3	1.4.4
librepo		1,14,5
libreport-filessystem		2,15,2
libseccomp		2.5.3
libselinux	2.1.10	3.4
libselinux-utils	2.1.10	3.4
libsemanage	2.1.6	3.4
libsepol	2.1.7	3.4
libsigsegv		2,13
libsmartcols	2,23,2	2,37,4
libsolv		0,7,22
libss	1,43,5	1,46,5
libssh2	1.4.2	
libstdc++		11.4.1
libstdc++72	7.2.1	
libsysfs	2.1.0	
libtasn1	2.3	4,19,0
libtextstyle		0,21
libudev	173	

Pacchetto	AL1 Minimo	AL2023 Minimo
libunistring	0,9,3	0,9,10
libuser	0,60	0,63
libutempter	1.1.5	1.2.1
libuuid	2,23,2	2,37,4
libverto	0,2,5	0,32
libxcrypt		4,4,33
libxml2	2,9,1	210.4
libyaml	0,16	02,5
libzstd		1,5,5
linux-firmware-whe nce		20210208
logrotate	3,78	3,20,1
lua	5.1.4	
lua-libs		5.4.4
lz4-libs		1.9.4
make	3,82	
man-db		2,9,3
microcode_ctl	2.1	2.1
mingetty	1,08	
mpfr		4.1.0
ncurses	5.7	6.2

Pacchetto	AL1 Minimo	AL2023 Minimo
ncurses-base	5.7	6.2
ncurses-libs	5.7	6.2
nettle		3.8
net-tools	1,60	2.0
newt	0,52,11	
newt-python27	0,52,11	
npth		1.6
nspr	4,25,0	
nss	3,53,1	
nss-pem	1.0.3	
nss-softokn	3,53,1	
nss-softokn-freebl	3,53,1	
nss-sysinit	3,53,1	
nss-tools	3,53,1	
nss-util	3,53,1	
ntp	4.2.8 p15	
ntpdate	4.2.8 p15	
numactl-libs		2.0,14
oniguruma		6,9,7,1
openldap	2,4,40	2,4,57

Pacchetto	AL1 Minimo	AL2023 Minimo
openssh	7,4p 1	8,7p1
openssh-clients		8,7p1
openssh-server	7,4p1	8,7p1
openssl	1,0,2k	3,0,8
openssl-lib		30,8
openssl-pkcs11		0,4,12
os-prober		1,77
p11-kit	0,18,5	0,24,1
p11-kit-trust	0,18,5	0,24,1
pam	1.1.8	1.5.1
passwd	0,79	0,80
pciutils	3,1,10	3.7.0
pciutils-lib	3,1,10	3.7.0
pcre	8,21	
pcre2		10,40
pcre2-syntax		10,40
pinentry	0,7,6	
pkgconfig	0,27,1	
policycoreutils	2,1,12	3.4
popt	1.13	1,18

Pacchetto	AL1 Minimo	AL2023 Minimo
procmail	3,22	
procps	32,8	
procps-ng		3,3,17
psmisc	22,20	23,4
pth	2.0.7	
publicsuffix-list-dafsa		20240212
python27	2,7,18	
python27-babel	0.9.4	
python27-backports	1.0	
python27-backports-ssl_match_hostname	3,40,2	
python27-chardet	2.0.1	
python27-configobj	4,7,2	
python27-iniparse	0,31	
python27-jinja2	27.2	
python27-jsonpatch	1.2	
python27-jsonpointer	1.0	
python27-libs	2,7,18	
python27-markupsafe	0,11	
python27-pycurl	7,19,0	

Pacchetto	AL1 Minimo	AL2023 Minimo
python27-pygments	0.3	
python27-pyliblzma	0,5,3	
python27-pyattr	0,50	
python27-PyYAML	3,10	
python27-requests	1.2.3	
python27-setuptools	36,27	
python27-six	1.8.0	
python27-urlgrabber	3,10	
python27-urllib3	1,24,3	
python3		3,9,16
python3-attrs		20,3,0
python3-audit		30.6
python3-awscrt		0,19,19
python3-babel		2,9,1
python3-cffi		1,14,5
python3-chardet		4.0.0
python3-colorama		04.4
python3-configobj		50.6
python3-cryptography		36,0
python3-dateutil		28.1

Pacchetto	AL1 Minimo	AL2023 Minimo
python3-dbus		1,2,18
python3-distro		1.5.0
python3-dnf		4.14.0
python3-dnf-plugins-core		4.3.0
python3-docutils		0,16
python3-gpg		1.15.1
python3-hawkey		0,69,0
python3-idna		(2.10)
python3-jinja2		211,3
python3-jmespath		0.10.0
python3-jsonpatch		1,21
python3-jsonpointer		2.0
python3-jjsonschema		3.2.0
python3-libcomps		01,20
python3-libdnf		0,69,0
python3-libs		3,9,16
python3-libselenium		3.4
python3-libsemanage		3.4
python3-markupsafe		1.1.1
python3-netifaces		0,10,6

Pacchetto	AL1 Minimo	AL2023 Minimo
python3-oauthlib		3.0.2
python3-pip-wheel		21,31
python3-ply		3,11
python3-policycore utils		3.4
python3-prettytable		0.7.2
python3-prompt-too lkit		3,0,24
python3-pycparser		2,20
python3-pyrsistent		0,17,3
python3-pyserial		3.4
python3-pysocks		1.7.1
python3-pytz		2022,7,1
python3-pyyaml		5.4.1
python3-requests		2,25,1
python3-rpm		4,161,3
python3-ruamel-yaml		0,16,6
python3-ruamel-yaml- clib		0,12
python3-setools		4.4.1
python3-setuptools		59,6,0

Pacchetto	AL1 Minimo	AL2023 Minimo
python3-setuptools-wheel		59,6,0
python3-six		1.15.0
python3-systemd		235
python3-urllib3		1,25,10
python3-wcwidth		0,2,5
readline	6.2	8.1
rng-tools		6,14
rootfiles	8.1	8.1
rpm	4.11.3	4,161,3
rpm-build-libs	411,3	4,161,3
rpm-libs	411,3	4,161,3
rpm-plugin-selinux		4,161,3
rpm-plugin-systemd-inhibit		4,161,3
rpm-python27	411,3	
rpm-sign-libs		4,161,3
rsyslog	5,8,10	
sbsigntools		0.9.4
sed	42,1	4.8
selinux-policy		381,45

Pacchetto	AL1 Minimo	AL2023 Minimo
selinux-policy-targeted		381,45
sendmail	8,14,4	
setserial	2,17	
setup	2,8,14	2,13,7
shadow-utils	4.1.4.2	4.9
shared-mime-info	1.1	
slang	2.2.1	
sqlite	3,7,17	
sqlite-libs		3,4,0
sudo	1,8,23	1,9,15
sysctl-defaults	1.0	1.0
sysfsutils	2.1.0	
systemd		252,23
systemd-libs		252,23
systemd-networkd		252,23
systemd-pam		252,23
systemd-resolved		252,23
systemd-udev		252,23
system-release	2018,03	2023,620241031
sysvinit	2,87	

Pacchetto	AL1 Minimo	AL2023 Minimo
tar	1,26	1,34
tcp_wrappers-libs	7.6	
tzdata	2023 c	2024a
udev	173	
update-motd	1.0.1	2.2
upstart	0,6,5	
userspace-rcu		0.12.1
ustr	1.0.4	
util-linux	2,23,2	2,37,4
util-linux-core		2,37,4
vim-data	9,0,2120	9,0,2153
vim-minimal	9,0,2120	9,0,2153
which	2,19	2,21
xfspgrog		5,18,0
xz	5.2.2	5.2.5
xz-libs	5.2.2	5.2.5
yum	3.4.3	4.14.0
yum-metadata-parser	1.1.4	
yum-plugin-priorities	1,1,31	

Pacchetto	AL1 Minimo	AL2023 Minimo
yum-plugin-upgrade-helper	1,1,31	
zlib	1.2.8	1,2,11
zram-generator		1.1.2
zram-generator-defaults		1.1.2
zstd		1,5,5

Confronto dei pacchetti installati sulle immagini dei container di base Amazon Linux 1 (AL1) e Amazon Linux 2023

Un confronto tra le immagini del contenitore di base RPMs attuali AL1 e quelle del contenitore di base AL2 023.

Pacchetto	AL1 Contenitore	AL2Contenitore 023
alternatives		1.15
amazon-linux-repo-cdn		2023,620241031
audit-libs		3,0,6
basesystem	10,0	11
bash	4,2,46	5,2,15
bzip2-libs	1.0.6	1.0.8
ca-certificates	2023,2,62	2023,2,68
chkconfig	1,349,3	

Pacchetto	AL1 Contenitore	AL2Contenitore 023
coreutils	8,22	
coreutils-single		8,32
crypto-policies		20220428
curl	7,61,1	
curl-minimal		8,5,0
cyrus-sasl-lib	2,1,23	
db4	4,7,25	
db4-utils	4,7,25	
dnf		4.14.0
dnf-data		4.14.0
elfutils-default-yama-scope		0.188
elfutils-libelf	0,168	0.188
elfutils-libs		0.188
expat	2.1.0	2.5.0
file-libs	5,37	5,39
filesystem	2,4,30	3,14
gawk	3.1.7	5.1.0
gdbm	1.8.0	
gdbm-libs		1,19
glib2	2,36,3	2,74,7

Pacchetto	AL1 Contenitore	AL2Contenitore 023
glibc	2,17	2,34
glibc-common	2,17	2,34
glibc-minimal-lang pack		2,34
gmp	6.0.0	62,1
gnupg2	2.0,28	
gnupg2-minimal		2.3.7
gpgme	1.4.3	1.15.1
grep	2,20	3.8
gzip	1.5	
info	5.1	
json-c		0,14
keyutils-libs	1.5.8	1.6.3
krb5-libs	1.15.1	1,21,3
libacl	2,2,49	2.3.1
libarchive		3,74
libassuan	2.0.3	2,5,5
libattr	2,4,46	2.5.1
libblkid		2,37,4
libcap	2,16	2,48
libcap-ng		0.8.2

Pacchetto	AL1 Contenitore	AL2Contenitore 023
libcom_err	1,43,5	1,46,5
libcomps		01,20
libcurl	7,61,1	
libcurl-minimal		8,5,0
libdnf		0,69,0
libffi	3,0,13	34.4
libgcc		114,1
libgcc72	7.2.1	
libgcrypt	1.5.3	1.10.2
libgomp		11.4.1
libgpg-error	1.11	1,42
libicu	50,2	
libidn2	2.3.0	2.3.2
libmodulemd		2.13.0
libmount		2,37,4
libnghttp2	1,33,0	1,59,0
libpsl	0.6.2	0,21,1
librepo		1,14,5
libreport-filessystem		2,15,2
libselinux	2.1.10	3.4

Pacchetto	AL1 Contenitore	AL2Contenitore 023
libsepol	2.1.7	3.4
libsigsegv		2,13
libsmartcols		2,37,4
libsolv		0,7,22
libssh2	1.4.2	
libstdc++		11,4,1
libstdc++72	7.2.1	
libtasn1	2.3	4,19,0
libunistring	0,9,3	0,9,10
libuuid		2,37,4
libverto	0,2,5	0,32
libxcrypt		4,4,33
libxml2	2,9,1	210.4
libxml2-python27	29.1	
libyaml		0,2,5
libzstd		1,5,5
lua	5.1.4	
lua-libs		5.4.4
lz4-libs		1.9.4
make	3,82	

Pacchetto	AL1 Contenitore	AL2Contenitore 023
mpfr		4.1.0
ncurses	5.7	
ncurses-base	5.7	6.2
ncurses-libs	5.7	6.2
npth		1.6
nspr	4,25,0	
nss	3,53,1	
nss-pem	1.0.3	
nss-softokn	3,53,1	
nss-softokn-freebl	3,53,1	
nss-sysinit	3,53,1	
nss-tools	3,53,1	
nss-util	3,53,1	
openldap	2,4,40	
openssl	1,0,2k	
openssl-libs		3,0,8
p11-kit	0,18,5	0,24,1
p11-kit-trust	0,18,5	0,24,1
pcre	8,21	
pcre2		10,40

Pacchetto	AL1 Contenitore	AL2Contenitore 023
pcresyntax		10,40
pinentry	0,7,6	
pkgconfig	0,27,1	
popt	1.13	1,18
pth	2.0.7	
publicsuffix-list-dafsa		20240212
python27	2,7,18	
python27-chardet	2.0.1	
python27-iniparse	0,31	
python27-kitchen	1.1.1	
python27-libs	2,7,18	
python27-pycurl	7,19,0	
python27-pygpme	0.3	
python27-pyliblzma	0,5,3	
python27-pyattr	0,5,0	
python27-urlgrabber	3,10	
python3		3,9,16
python3-dnf		4.14.0
python3-gpg		1.15.1
python3-hawkey		0,69,0

Pacchetto	AL1 Contenitore	AL2Contenitore 023
python3-libcomps		01,20
python3-libdnf		0,69,0
python3-libs		3,9,16
python3-pip-wheel		21,31
python3-rpm		4,16,1,3
python3-setuptools-wheel		59,60
readline	6.2	8.1
rpm	411,3	4,161,3
rpm-build-libs	411,3	4,161,3
rpm-libs	411,3	4,161,3
rpm-python27	411,3	
rpm-sign-libs		4,161,3
sed	42,1	4.8
setup	2,8,14	2,13,7
shared-mime-info	1.1	
sqlite	3,7,17	
sqlite-libs		3,4,0
sysctl-defaults	1.0	
system-release	2018,03	2023,620241031
tar	1,26	

Pacchetto	AL1 Contenitore	AL2Contenitore 023
tzdata	2023 c	2024a
xz-libs	5.2.2	5.2.5
yum	3.4.3	4.14.0
yum-metadata-parser	1.1.4	
yum-plugin-ovl	1,1,31	
yum-plugin-priorities	1,1,31	
yum-utils	1,1,31	
zlib	1.2.8	1,2,11

AL2023 requisiti di sistema

Questa sezione descrive i requisiti di sistema per l'utilizzo di AL2 023.

Argomenti

- [Requisiti della CPU per l'esecuzione di 023 AL2](#)
- [Requisiti di memoria \(RAM\) per l'esecuzione di 023 AL2](#)

Requisiti della CPU per l'esecuzione di 023 AL2

Per eseguire qualsiasi codice AL2 023, il processore utilizzato deve soddisfare determinati requisiti minimi. I tentativi di eseguire AL2 023 su un CPUs computer che non soddisfa questi requisiti potrebbero causare errori di istruzione illegali nelle prime fasi dell'esecuzione del codice.

I requisiti minimi si applicano a [AL2023 su Amazon EC2AL2023 in contenitori](#), e [AL2023 al di fuori di Amazon EC2](#).

Requisiti della CPU ARM per AL2 023

Tutte le AL2 023 aarch64 (ARM) i binari sono creati per 64 bit. Nessun 32 bit ARM sono disponibili file binari, quindi a 64 bit ARM È richiesta la CPU.

Note

Per le istanze basate su ARM, AL2 023 supporta solo i tipi di istanza che utilizzano processori Graviton2 o successivi. AL2023 non supporta le istanze A1.

AL2023 richiede un processore conforme alla ARMv8 versione 2.2 con Cryptography Extension (). ARMv8.2+crypto Tutti i pacchetti AL2 023 per aarch64 sono compilati con il flag del compilatore. -march=armv8.2-a+crypto Tuttavia, cerchiamo di stampare messaggi di errore corretti quando si tenta di eseguire il codice AL2 023 su versioni precedenti ARM processori, è possibile che il primo messaggio di errore sia un errore di istruzione illegale.

Note

A causa dei requisiti di `aarch64` base della CPU AL2 023, tutti Raspberry Pi sistemi precedenti al Raspberry Pi 5 non soddisfano i requisiti minimi della CPU.

Requisiti della CPU x86-64 per 023 AL2

Tutti i `x86-64` file binari AL2 023 sono creati per la `x86-64v2` revisione dell'architettura e vengono passati al `x86-64` compilatore. `-march=x86-64-v2`

La `x86-64v2` revisione dell'architettura aggiunge le seguenti funzionalità della CPU all'architettura di base: `x86-64`

- `CMPXCHG16B`
- `LAHF-SAHF`
- `POPCNT`
- `SSE3`
- `SSE4_1`
- `SSE4_2`
- `SSSE3`

Ciò corrisponde approssimativamente ai `x86-64` processori rilasciati nel 2009 o successivamente. Gli esempi includono Intel Nehalem, AMD Jaguar, Atom Silvermont, insieme a VIA Nano e Eden C microarchitetture.

In Amazon EC2, sono supportati tutti i tipi di `x86-64` istanze `x86-64v2 M1C1`, comprese le famiglie di M2 istanze.

Nessun `x86` a 32 bit (`i686`) Vengono compilati AL2 023 binari. Sebbene AL2 023 mantenga il supporto per l'esecuzione di file binari dello spazio utente a 32 bit, questa funzionalità è obsoleta e potrebbe essere rimossa in una futura versione principale di Amazon Linux. Per ulteriori informazioni, consulta [Pacchetti x86 \(i686\) a 32 bit](#).

Requisiti di memoria (RAM) per l'esecuzione di 023 AL2

La EC2 .nano famiglia di tipi di istanze Amazon (t2.nano, t3.nanot3a.nano, et4g.nano) dispone di 512 MB di RAM, che è il requisito minimo per AL2 023.

Note

Sebbene il requisito minimo sia 512 MB, questi tipi di istanze hanno limiti di memoria e funzionalità e prestazioni potrebbero essere limitate.

AL2023 immagini non sono state testate su sistemi con meno di 512 MB di RAM. L'esecuzione di immagini di container basate su AL2 023 in meno di 512 MB di RAM dipenderà dal carico di lavoro containerizzato.

Alcuni carichi di lavoro, ad esempio `dnf upgrade` tra alcune versioni AL2 023, possono richiedere più di 512 MB di RAM. Per questo motivo, la versione [AL2023.3](#) ha introdotto l'abilitazione per impostazione `zram` predefinita per le istanze con meno di 800 MB di RAM. Per i carichi di lavoro containerizzati, ciò significa che alcuni carichi di lavoro potrebbero funzionare correttamente su AL2 023 istanze con questa quantità di memoria, ma fallire se eseguiti in un contenitore con un utilizzo limitato a questa quantità di memoria.

Per i tipi di istanze con meno di 800 MB di RAM, AL2 023 (a partire dalla versione [AL2023.3](#) o successiva) abiliterà lo scambio basato per impostazione predefinita.

`zram` Esempi di tipi di EC2 istanze Amazon con meno di 800 MB di memoria includono `t4g.nanot3a.nano`, `t3.nano`, `t2.nano`, `et1.micro`. Ciò significa un minor numero di scenari di esaurimento della memoria per questi tipi di istanze, perché AL2 023 comprimerà e decomprimerà su richiesta le pagine di memoria. In questo modo, vengono abilitati i carichi di lavoro che altrimenti richiederebbero un tipo di istanza con più memoria, a scapito dell'utilizzo della CPU che serve per eseguire la compressione.

Il kernel Amazon Linux

Amazon Linux 2023 (AL2023) rilascia un nuovo kernel Long-Term Support (LTS) nel primo trimestre di ogni anno, basato sul kernel LTS annuale della comunità Linux upstream. Ogni nuovo kernel LTS offre le ultime correzioni di sicurezza, miglioramenti delle prestazioni, supporto hardware e funzionalità del kernel Linux upstream.

Ciclo di vita del kernel

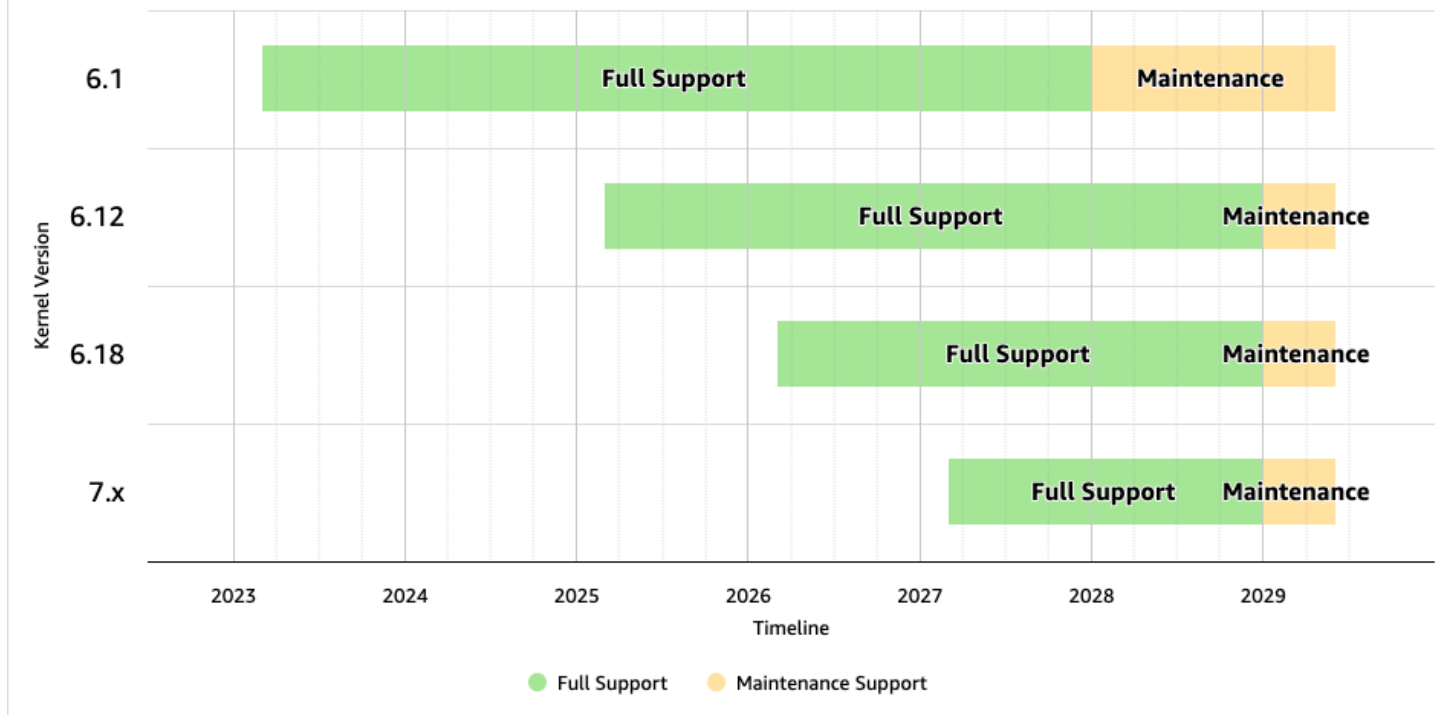
Ogni kernel AL2023 LTS è supportato per quattro anni in due fasi:

1. Supporto completo (anni 1—2): il kernel riceve correzioni per tutte le severità CVE tramite ribasi regolari sul kernel LTS upstream. Questa fase si allinea alla finestra di supporto LTS della comunità Linux upstream. Se upstream estende la finestra di supporto LTS, Amazon Linux seguirà l'esempio.
2. Supporto di manutenzione (anni 3-4): dopo la fine del periodo di supporto LTS upstream, il team di Amazon Linux continua a eseguire il backport principalmente delle correzioni critiche e importanti CVEs (punteggio CVSS 7.0 e superiore), nonché delle vulnerabilità sfruttate note. In questa fase non viene effettuato il backport di gravità bassa e media. CVEs

Dopo quattro anni, il kernel raggiunge la fine del ciclo di vita e non riceve più aggiornamenti di sicurezza. I kernel più vecchi rimangono disponibili tramite i repository ed è possibile continuare a utilizzarli. Non dovreste aspettarvi ulteriori patch o correzioni. Si consiglia di effettuare l'aggiornamento a un kernel supportato prima di questa data. I kernel specifici non AMIs verranno più aggiornati dopo la data di fine del supporto di un kernel.

Il seguente grafico mostra la tempistica di supporto per gli attuali kernel Amazon Linux LTS:

Amazon Linux 2023 Kernel Release Timeline



Aggiornamenti del kernel

A partire da giugno 2026, AL2023 aggiornerà il kernel predefinito ogni anno. Il [al2023-ami-kernel-default](#) set di AMIs verrà aggiornato all'ultimo kernel LTS, in modo che le istanze appena lanciate forniscano la nuova versione del kernel.

Le istanze in esecuzione non vengono aggiornate automaticamente a un nuovo kernel. Per aggiornare il kernel su un'istanza esistente, è necessario installare esplicitamente il nuovo pacchetto kernel e riavviare. Per informazioni dettagliate, vedi [Aggiornamento del kernel Linux su AL2023](#).

Consigliamo vivamente di adottare tempestivamente nuovi kernel per beneficiare dei più recenti miglioramenti in termini di sicurezza e prestazioni. I kernel più vecchi ricevono meno correzioni di sicurezza e sono più lente nel tempo. Incorpora gli aggiornamenti del kernel nelle pipeline di test e distribuzione esistenti per convalidare la compatibilità prima del lancio in produzione.

Le versioni del kernel si sovrappongono tra le distribuzioni

Per semplificare le migrazioni tra le distribuzioni Amazon Linux, sarà disponibile almeno una versione del kernel sia sulla distribuzione Amazon Linux attuale che su quella successiva. Ciò ti consente di

eseguire prima l'aggiornamento a un nuovo kernel della distribuzione esistente, convalidare i carichi di lavoro e quindi migrare alla nuova distribuzione con la certezza che la stessa versione del kernel sia disponibile lì.

Gestione CVE

AL2023 indirizza il kernel CVEs come segue:

- Le vulnerabilità sfruttate critiche e importanti CVEs (CVSS 7.0 e versioni successive) e note vengono trasferite su tutti i kernel supportati.
- Le basse e le medie CVEs (CVSS inferiore a 7.0) vengono risolte tramite regolari ribasi LTS upstream durante la fase di supporto completo. Durante la fase di supporto alla manutenzione, questi non vengono sottoposti a backport. CVEs Se un CVE basso o medio non viene risolto da un rebase LTS upstream entro 60 giorni, verrà contrassegnato come «Nessuna correzione pianificata» in [Amazon Linux Security Center](#).

L'esecuzione del kernel più recente disponibile è il modo migliore per ottenere aggiornamenti recenti di sicurezza, prestazioni e funzionalità.

Cosa dovresti fare

1. Implementa l'integrazione continua (CI) per le tue applicazioni: prima di apportare modifiche alla configurazione di produzione, assicurati che sia correttamente convalidata in una fase di test. Includi gli aggiornamenti del kernel nella convalida.
2. Resta aggiornato al kernel più recente: adotta ogni nuovo kernel LTS annuale non appena è disponibile. I kernel più recenti ricevono le correzioni di sicurezza più velocemente. Consuma la [al2023-ami-kernel-default](#) serie di AMIs per accedere automaticamente alla versione del kernel consigliata dal team di Amazon Linux.
3. Automatizza gli aggiornamenti: utilizza strumenti come [AWS Systems Manager](#) per gestire gli aggiornamenti del kernel in tutta la tua flotta.
4. Pianifica i riavvii: ogni aggiornamento del kernel richiede un riavvio. Inserisci le finestre di riavvio nelle tue pianificazioni di manutenzione.

Driver NVIDIA

Amazon Linux 2023 fornisce driver GPU NVIDIA e pacchetti di toolkit CUDA tramite un repository dedicato. Questo repository è gestito AWS e fornisce avvisi di sicurezza tramite [Amazon Linux Security Center \(ALAS\)](#).

Argomenti

- [Informazioni sul repository NVIDIA](#)
- [Abilitazione del repository NVIDIA](#)
- [Installazione dei driver NVIDIA](#)
- [Installazione del toolkit CUDA](#)
- [Rimozione del repository NVIDIA](#)

Informazioni sul repository NVIDIA

Il repository AL2023 NVIDIA rispecchia i pacchetti del repository [ufficiale NVIDIA CUDA per](#). AL2023 AWS qualifica il software NVIDIA con AL2023 release candidate prima della redistribuzione e fornisce avvisi di sicurezza per i pacchetti contenuti in questo repository.

L'archivio è disponibile in tutte le regioni AWS commerciali, incluse le regioni AWS GovCloud (Stati Uniti) e le regioni della AWS Cina.

Il repository fornisce driver NVIDIA Tesla (elaborazione per data center) e driver grafici per architetture x86_64. I driver GRID, utilizzati per le funzionalità di visualizzazione virtuale e workstation remote, non sono inclusi. Per l'installazione dei driver GRID, consulta [Installare i driver NVIDIA](#) nella Guida per l'utente EC2.

Abilitazione del repository NVIDIA

Per abilitare il repository NVIDIA sulla tua AL2023 istanza, installa il pacchetto. `nvidia-release`. Questo aggiunge la configurazione del repository e le chiavi GPG al sistema.

```
[ec2-user ~]$ sudo dnf install nvidia-release -y
```

Verifica che il repository sia stato aggiunto:

```
[ec2-user ~]$ dnf repolist
```

Dovresti vedere il `amazonlinux-nvidia` repository nell'elenco.

```
repo id                repo name
status
amazonlinux            Amazon Linux 2023 repository
enabled
amazonlinux-nvidia    Amazon Linux 2023 NVIDIA repository
enabled
```

Installazione dei driver NVIDIA

Dopo aver abilitato il repository, è possibile installare i pacchetti driver NVIDIA utilizzando `dnf`

1. Installa gli header del kernel e i pacchetti di sviluppo per il kernel in esecuzione:

```
[ec2-user ~]$ sudo dnf install kernel-devel-$(uname -r) kernel-headers-$(uname -r) -y
```

2. Installa il driver NVIDIA:

```
[ec2-user ~]$ sudo dnf install nvidia-driver-cuda -y
```

3. Riavvia l'istanza:

```
[ec2-user ~]$ sudo reboot
```

4. Dopo il riavvio, verifica che il driver sia caricato:

```
[ec2-user ~]$ nvidia-smi
```

Installazione del toolkit CUDA

Dopo aver installato il driver NVIDIA, puoi installare il toolkit CUDA:

```
[ec2-user ~]$ sudo dnf install cuda-toolkit -y
```

Note

Per le istanze GPU che richiedono NVIDIA Fabric Manager (come i tipi di istanze P4d, P5 e P6), installa e abilita i pacchetti aggiuntivi:

```
[ec2-user ~]$ DRV_BRANCH="$(modinfo nvidia | grep "^version:" | tr -s ' ' | cut  
-d ' ' -f 2)"  
[ec2-user ~]$ sudo dnf install nvidia-fabricmanager-${DRV_BRANCH} -y  
[ec2-user ~]$ sudo systemctl enable --now nvidia-fabricmanager  
[ec2-user ~]$ sudo systemctl enable --now nvidia-persistenced
```

Verifica che Fabric Manager sia in esecuzione e che siano connessi tramite: GPUs NVSwitch

```
[ec2-user ~]$ sudo systemctl status nvidia-fabricmanager  
[ec2-user ~]$ nvidia-smi topo -m
```

Nella matrice della topologia, le connessioni tra di loro GPUs dovrebbero mostrare NV dei collegamenti, a indicare la NVSwitch connettività attiva.

Per istruzioni dettagliate sull'installazione dei driver NVIDIA sulle istanze GPU EC2, inclusi i requisiti specifici del tipo di istanza, consulta [Installare i driver pubblici NVIDIA](#) nella Guida per l'utente di EC2.

Rimozione del repository NVIDIA

Per rimuovere la configurazione del repository NVIDIA dal sistema:

```
[ec2-user ~]$ sudo dnf remove nvidia-release -y
```

Important

La rimozione della configurazione del repository non rimuove i pacchetti NVIDIA già installati sul sistema.

AL2Desktop grafico 023

Amazon Linux 2023 fornisce un'interfaccia grafica opzionale, leggera e ottimizzata per il cloud basata su GNOME a partire dalla versione 2023.7. Questo moderno ambiente desktop offre funzionalità di produttività avanzate con strumenti integrati come Firefox per una navigazione sicura, pur mantenendo il supporto di Amazon DCV e VNC per l'accesso remoto.

Note

AL2023 segue da vicino la versione upstream di Firefox Extended Support Release (ESR) e si aggiorna alla prossima ESR non appena disponibile. [Per ulteriori informazioni, consulta il calendario di rilascio di Firefox ESR e le note di rilascio di Firefox.](#)

Argomenti correlati

Per ulteriori informazioni sull'installazione dell'ambiente desktop grafico, consulta la seguente documentazione:

- [Tutorial: Installare l'ambiente desktop GNOME su AL2023](#)

Pacchetti supplementari per Amazon Linux

Questa sezione introduce i pacchetti supplementari per Amazon Linux (SPAL) e ne descrive i vantaggi e i limiti, insieme alle linee guida per la segnalazione di problemi relativi ai pacchetti.

Argomenti

- [Cosa sono i pacchetti supplementari per Amazon Linux \(o SPAL\)?](#)
- [Vantaggi](#)
- [Support dei pacchetti SPAL](#)
- [Segnalazione di problemi relativi ai pacchetti](#)
- [Argomenti correlati](#)
- [Tutorial: Configurare l'archivio SPAL su AL2023](#)

Cosa sono i pacchetti supplementari per Amazon Linux (o SPAL)?

Supplementary Packages for Amazon Linux (SPAL) è un repository di pacchetti dedicato che fornisce l'accesso a migliaia di pacchetti aggiuntivi derivati da [Extra Packages for Enterprise Linux 9](#) (). EPEL9. Questi pacchetti completano il software esistente disponibile nel core di Amazon Linux 2023.

SPAL semplifica la distribuzione del software fornendo pacchetti predefiniti compatibili con AL2 023, eliminando la necessità per i clienti di creare autonomamente i pacchetti a partire dal codice sorgente. Ciò consente di risparmiare tempo e fatica nel processo di installazione del software.

Note

SPAL è disponibile in tutte le regioni AWS commerciali, incluse le regioni AWS GovCloud (Stati Uniti) e la Cina, per AL2 023 istanze con una versione di rilascio di 2023.9.20251117 o successiva.

Vantaggi

SPAL offre diversi vantaggi chiave agli utenti di Amazon Linux 2023:

- Ampliamento dei casi d'uso di AL2 023: l'accesso ai più diffusi pacchetti aggiuntivi oltre al repository principale AL2 023, come, `GDAL` or `pandocdrbd-utils`, consente ai clienti di supportare molteplici esigenze aziendali e di sviluppo.
- Semplificazione della gestione dei pacchetti AL2 023: fornendo pacchetti precompilati per AL2 023, viene eliminato il processo di creazione di pacchetti aggiuntivi dal codice sorgente, risparmiando tempo e riducendo il rischio di errori di compilazione.
- Semplificazione della migrazione AL2 alla versione AL2 023: l'archivio SPAL consente la migrazione senza interruzioni dei carichi di lavoro da 0 a 023, compresi i carichi di lavoro che dipendono dai pacchetti. AL2 AL2 EPEL7

Support dei pacchetti SPAL

I pacchetti SPAL non sono coperti dallo stesso livello di supporto dei pacchetti core AL2 023, che ricevono supporto per l'intera durata di Amazon Linux 2023.

Important

Prima di utilizzare SPAL, i clienti devono valutare attentamente le seguenti considerazioni:

- I pacchetti SPAL NON sono coperti dai piani di AWS supporto.
- I pacchetti SPAL vengono forniti «così come sono» dall'upstream. EPEL9
- I pacchetti SPAL NON riceveranno il tracciamento di sicurezza CVE. AWS
- I pacchetti SPAL ricevono patch di sicurezza e correzioni di bug esclusivamente dall'upstream, se disponibili. EPEL9

Segnalazione di problemi relativi ai pacchetti

Se riscontrate un problema con un pacchetto SPAL, vi consigliamo di verificare innanzitutto se lo stesso problema si verifica con il pacchetto corrispondente nel repository upstream. EPEL9 Per questo, consulta l'[elenco dei problemi](#) sull'archivio EPEL upstream.

Se il problema non è presente in EPEL9, crea un problema nel [GitHubrepository Amazon Linux 2023](#), poiché ciò indica che il problema è specifico della build o della configurazione del pacchetto SPAL.

Questo approccio garantisce che i problemi vengano risolti dai manutentori appropriati e contribuisce alla qualità complessiva dei pacchetti SPAL e EPEL upstream.

Note

I problemi segnalati verranno gestiti con la massima diligenza.

Argomenti correlati

Per informazioni sulla configurazione di SPAL sul sistema, consultate la seguente pagina di documentazione:

- [Tutorial: Configurare l'archivio SPAL su AL2023](#)

Tutorial: Configurare l'archivio SPAL su AL2023

Supplementary Packages for Amazon Linux (SPAL) è un archivio di pacchetti aggiuntivo AL2023 che fornisce ai clienti l'accesso a migliaia di pacchetti open source.

Il seguente tutorial ti aiuta a configurare il repository SPAL sulla tua istanza. AL2023 Installando il repository, avrete accesso a tutti i pacchetti RPM disponibili in SPAL. Una volta installati, potete usare il vostro gestore di pacchetti per installare e usare questi pacchetti sul vostro sistema.

Indice

- [Prerequisiti](#)
- [Verifica dei prerequisiti](#)
- [Installazione di SPAL sul sistema](#)
- [Installazione dei pacchetti SPAL](#)
- [Scaricamento dei pacchetti sorgente SPAL](#)
- [Installazione dei pacchetti SPAL debuginfo](#)
- [Disinstallazione del repository SPAL dal sistema](#)
- [Argomenti correlati](#)

Prerequisiti

Questo tutorial presuppone che tu abbia già avviato un'istanza utilizzando la versione di AL2023 rilascio 2023.9.20251117 o successiva. Per ulteriori informazioni, consulta le [Aggiornamenti AL2023](#) pagine [AL2023 su Amazon EC2](#) and.

Verifica dei prerequisiti

- Per verificare che l'istanza soddisfi i prerequisiti, puoi controllare la versione `system-release` installata sul tuo sistema.

Per verificare la versione del pacchetto, puoi usare il seguente comando.

```
[ec2-user ~]$ rpm -qi system-release
```

Il comando mostrerà informazioni sul pacchetto, inclusa la versione principale.

```
Name       : system-release
Version    : 2023.9.20251117
...
```

Note

Assicurati di avere `system-release` installato la versione più recente di. Puoi eseguire `sudo dnf upgrade` l'aggiornamento alla versione più recente.

Installazione di SPAL sul sistema

1. Installa il `spal-release` pacchetto sul tuo sistema. Questo aggiunge il file `.repo` di configurazione e le chiavi GPG al sistema.

```
[ec2-user ~]$ sudo dnf install spal-release
```

Note

Durante l'installazione, verrà visualizzata la dichiarazione di supporto. La dichiarazione spiega l'ambito di supporto e le limitazioni di SPAL. Vi preghiamo di dedicare del tempo a esaminare attentamente queste informazioni.

2. Verificate che la configurazione del repository SPAL sia stata aggiunta correttamente al sistema.

```
[ec2-user ~]$ cat /etc/yum.repos.d/amazonlinux-spal.repo
```

Dovreste vedere i tre repository configurati sul vostro sistema: `amazonlinux-spal`, `amazonlinux-spal-source` e `amazonlinux-spal-debuginfo`

Puoi anche controllare l'elenco dei repository configurati eseguendo `dnf repolist`

```
[ec2-user ~]$ dnf repolist --all
```

Note

Il `--all` flag è necessario per visualizzare sia i repository abilitati che quelli disabilitati.

Tutti e tre i repository SPAL dovrebbero essere disponibili. Tieni presente che il repository Amazon Linux 2023 SPAL - Pacchetti sorgente e il repository Amazon Linux 2023 SPAL - I repository di debug sono disabilitati per impostazione predefinita.

```
repo id                repo name
  status
amazonlinux-spal       Amazon Linux 2023 SPAL repository
  enabled
amazonlinux-spal-source Amazon Linux 2023 SPAL repository - Source packages
  disabled
amazonlinux-spal-debuginfo Amazon Linux 2023 SPAL repository - Debug
  disabled
```

Installazione dei pacchetti SPAL

- Installa i pacchetti SPAL sul tuo sistema `dnf install` eseguendo il comando.

```
[ec2-user ~]$ sudo dnf install package
```

Note

È possibile `dnf list` utilizzarlo per visualizzare un elenco completo dei pacchetti SPAL.

```
[ec2-user ~]$ dnf list --repo=amazonlinux-spal
```

Note

SPAL è un repository con versioni. Assicurati di avere `system-release` installato la versione più recente di per vedere l'elenco più recente dei pacchetti.

Per ulteriori informazioni sugli aggiornamenti deterministici, puoi consultare [Aggiornamenti deterministici tramite repository con versioni su AL2023](#)

Scaricamento dei pacchetti sorgente SPAL

I pacchetti sorgente RPM (SRPM) vengono utilizzati principalmente dagli sviluppatori per creare pacchetti a partire dai sorgenti. Il repository dei sorgenti SPAL è disabilitato per impostazione predefinita. DNF lo abilita automaticamente quando si utilizzano comandi che richiedono pacchetti sorgente.

Per scaricare l'RPM sorgente per un pacchetto SPAL, esegui il seguente comando:

```
[ec2-user ~]$ dnf download --source package
```

Installazione dei pacchetti SPAL debuginfo

Il repository `debuginfo` contiene sia i pacchetti `debuginfo` che `debugsource`. Questi pacchetti forniscono simboli di debug e file sorgente utili per il debug e la profilazione delle applicazioni.

Il repository SPAL debuginfo è disabilitato per impostazione predefinita. DNF lo abilita automaticamente quando si utilizzano comandi che richiedono pacchetti di debug.

Per installare i pacchetti debuginfo e debugsource per un pacchetto SPAL, esegui il seguente comando:

```
[ec2-user ~]$ sudo dnf debuginfo-install package
```

Disinstallazione del repository SPAL dal sistema

1. Rimuovere la configurazione del repository SPAL utilizzando il comando. `dnf remove`

```
[ec2-user ~]$ sudo dnf remove spal-release
```

2. Verifica che il repository sia stato rimosso eseguendo il comando. `dnf repolist`

```
[ec2-user ~]$ dnf repolist
```

Important

La rimozione della configurazione del repository SPAL dal sistema non rimuove alcun pacchetto SPAL installato sul sistema.

Argomenti correlati

Per ulteriori informazioni sul repository Supplementary Packages for Amazon Linux, consulta la seguente documentazione:

- [Pacchetti supplementari per Amazon Linux](#)

Esecuzione di applicazioni su AL2 023

Questa sezione illustra i metodi per eseguire applicazioni su Amazon Linux 2023 (AL2023), inclusa la gestione dell'avvio (e del riavvio) e il controllo dell'utilizzo delle risorse.

Argomenti

- [Limitazione dell'utilizzo delle risorse di processo in AL2 023 utilizzando systemd](#)
- [Limitazione dell'utilizzo delle risorse di processo in AL2 023 utilizzando cgroups](#)

Limitazione dell'utilizzo delle risorse di processo in AL2 023 utilizzando systemd

Su Amazon Linux 2023 (AL2023), consigliamo di utilizzare `systemd` per controllare quali risorse possono essere utilizzate da processi o gruppi di processi. `systemd` è un sostituto potente e facile da usare della manipolazione `cgroups` manuale o dell'utilizzo di utilità come [`cpulimit`](#), in precedenza, disponibili solo per Amazon Linux nel repository di terze parti. [EPEL](#)

[Per informazioni complete, consulta la `systemd` documentazione originale di `systemd.resource-control` o il man pagina relativa a un'istanza 023. `systemd.resource-control` AL2](#)

Gli esempi seguenti utilizzeranno lo stress test della `stress-ng` CPU (`stress-ng` incluso nel pacchetto) per simulare un'applicazione che richiede un uso intensivo della CPU e `memcached` per simulare un'applicazione che utilizza molta memoria.

Gli esempi seguenti riguardano l'imposizione di un limite di CPU su un comando singolo e un limite di memoria su un servizio. La maggior parte dei vincoli di risorse `systemd` offerti può essere utilizzata in qualsiasi luogo in cui `systemd` venga eseguito un processo ed è possibile utilizzarne più di uno contemporaneamente. Gli esempi seguenti sono limitati a un singolo vincolo a scopo illustrativo.

Controllo delle risorse con esecuzione di comandi `systemd-run` singoli

Sebbene comunemente associato ai servizi di sistema, `systemd` può essere utilizzato anche da utenti non root per eseguire servizi, pianificare timer o eseguire processi una tantum. Nel seguente esempio, useremo `stress-ng` come applicazione di esempio. Nel primo esempio, lo eseguiremo utilizzando `systemd-run` l'account `ec2-user` predefinito e nel secondo esempio imporremo dei limiti all'utilizzo della CPU.

Example Utilizzalo **systemd-run** sulla riga di comando per eseguire un processo, senza limitare l'utilizzo delle risorse

1. Assicurati che il `stress-ng` pacchetto sia installato, poiché lo useremo per il nostro esempio.

```
[ec2-user ~]$ sudo dnf install -y stress-ng
```

2. Utilizzalo `systemd-run` per eseguire uno stress test della CPU di 10 secondi senza limitare la quantità di CPU che può utilizzare.

```
[ec2-user ~]$ systemd-run --user --tty --wait --property=CPUAccounting=1 stress-ng
--cpu 1 --timeout 10
Running as unit: run-u6.service
Press ^] three times within 1s to disconnect TTY.
stress-ng: info: [339368] setting to a 10 second run per stressor
stress-ng: info: [339368] dispatching hogs: 1 cpu
stress-ng: info: [339368] successful run completed in 10.00s
Finished with result: success
Main processes terminated with: code=exited/status=0
Service runtime: 10.068s
CPU time consumed: 9.060s
```

L'`--user` opzione indica `systemd-run` di eseguire il comando come utente con cui abbiamo effettuato l'accesso, l'`--tty` opzione indica a TTY è allegato, `--wait` significa attendere il completamento del servizio e l'`--property=CPUAccounting=1` opzione indica di registrare la quantità `systemd-run` di tempo della CPU utilizzata nell'esecuzione del processo.

L'opzione della riga di `--property` comando può essere utilizzata per passare `systemd-run` impostazioni che possono essere configurate in un file di `systemd.unit` configurazione.

Quando viene richiesto di caricare la CPU, il `stress-ng` programma utilizzerà tutto il tempo di CPU disponibile per eseguire il test per la durata richiesta per l'esecuzione. Per un'applicazione reale, può essere opportuno porre un limite al tempo di esecuzione totale di un processo. Nell'esempio seguente, chiederemo di eseguire l'operazione per un periodo più lungo rispetto `stress-ng` alla limitazione della durata massima che imponiamo al suo utilizzo. `systemd-run`

Example Utilizzalo **systemd-run** sulla riga di comando per eseguire un processo, limitando l'utilizzo della CPU a 1 secondo

1. Assicurati che `stress-ng` sia installato per eseguire questo esempio.

2. La `LimitCPU` proprietà è l'equivalente della `ulimit -t` quale limiterà la quantità massima di tempo sulla CPU che questo processo potrà utilizzare. In questo caso, poiché chiediamo un'esecuzione di `stress` di 10 secondi e limitiamo l'utilizzo della CPU a 1 secondo, il comando riceverà un `SIGXCPU` segnale e fallirà.

```
[ec2-user ~]$ systemd-run --user --tty --wait --property=CPUAccounting=1 --
property=LimitCPU=1 stress-ng --cpu 1 --timeout 10
Running as unit: run-u12.service
Press ^] three times within 1s to disconnect TTY.
stress-ng: info: [340349] setting to a 10 second run per stressor
stress-ng: info: [340349] dispatching hogs: 1 cpu
stress-ng: fail: [340349] cpu instance 0 corrupted bogo-ops counter, 1370 vs 0
stress-ng: fail: [340349] cpu instance 0 hash error in bogo-ops counter and run
flag, 3250129726 vs 0
stress-ng: fail: [340349] metrics-check: stressor metrics corrupted, data is
compromised
stress-ng: info: [340349] unsuccessful run completed in 1.14s
Finished with result: exit-code
Main processes terminated with: code=exited/status=2
Service runtime: 1.201s
CPU time consumed: 1.008s
```

Più comunemente, potresti voler limitare la percentuale di tempo della CPU che può essere consumata da un particolare processo. Nell'esempio seguente, limiteremo la percentuale di tempo di CPU che può essere consumato da `stress-ng`. Per un servizio reale, può essere opportuno limitare la percentuale massima di tempo di CPU che un processo in background può consumare per lasciare libere risorse per il processo che serve le richieste degli utenti.

Example Consente `systemd-run` di limitare un processo al 10% del tempo di CPU su una sola CPU

1. Assicurati che `stress-ng` sia installato per eseguire questo esempio.
2. Useremo la `CPUQuota` proprietà per indicare di `systemd-run` limitare l'utilizzo della CPU per il comando che stiamo per eseguire. Non stiamo limitando la quantità di tempo per cui il processo può essere eseguito, ma solo la quantità di CPU che può utilizzare.

```
[ec2-user ~]$ systemd-run --user --tty --wait --property=CPUAccounting=1 --
property=CPUQuota=10% stress-ng --cpu 1 --timeout 10
Running as unit: run-u13.service
Press ^] three times within 1s to disconnect TTY.
```

```
stress-ng: info: [340664] setting to a 10 second run per stressor
stress-ng: info: [340664] dispatching hogs: 1 cpu
stress-ng: info: [340664] successful run completed in 10.08s
Finished with result: success
Main processes terminated with: code=exited/status=0
Service runtime: 10.140s
CPU time consumed: 1.014s
```

Nota come CPU la contabilità ci dice che mentre il servizio è stato eseguito per 10 secondi, ha consumato solo 1 secondo del tempo effettivo della CPU.

Esistono molti modi di configurazione per `systemd` limitare l'utilizzo delle risorse per CPU, memoria, rete e IO. Consultate la `systemd` documentazione originale di [systemd.resource-control](#) o man pagina relativa a un'istanza 023 per una `systemd.resource-control` documentazione completa. AL2

Dietro le quinte, `systemd` si utilizzano funzionalità del kernel Linux che `cgroups` consentono di implementare questi limiti evitando la necessità di configurarli manualmente. La [documentazione del kernel Linux per cgroup-v2](#) contiene dettagli dettagliati sul `cgroups` lavoro.

Controllo delle risorse in un servizio **systemd**

È possibile aggiungere diversi parametri alla `[Service]` sezione dei `systemd` servizi per controllare l'utilizzo delle risorse di sistema. Questi includono limiti rigidi e morbidi. Per il comportamento esatto di ciascuna opzione, consulta la `systemd` documentazione originale di [systemd.resource-control](#) o man pagina per un'istanza 023. `systemd.resource-control` AL2

I limiti più comunemente usati `MemoryHigh` consistono nello specificare un limite di limitazione all'utilizzo della memoria, nell'impostare un limite massimo rigido (`MemoryMax` al quale, una volta raggiunto, viene richiamato l'OOM Killer) e `CPUQuota` (come illustrato nella sezione precedente). È anche possibile configurare pesi e priorità anziché numeri fissi.

Example Utilizzato **systemd** per impostare i limiti di utilizzo della memoria sui servizi

In questo esempio imposteremo un limite di utilizzo della memoria rigida per `memcached` una semplice cache chiave-valore e mostreremo come viene richiamato l'OOM Killer per quel servizio anziché per l'intero sistema.

1. Innanzitutto, dobbiamo installare i pacchetti richiesti per questo esempio.

```
[ec2-user ~]$ sudo dnf install -y memcached libmemcached-awesome-tools
```

2. Abilita `memcached.service` e poi avvia il servizio in modo che `memcached` sia in esecuzione.

```
[ec2-user ~]$ sudo systemctl enable memcached.service
Created symlink /etc/systemd/system/multi-user.target.wants/memcached.service # /usr/lib/systemd/system/memcached.service.
[ec2-user ~]$ sudo systemctl start memcached.service
```

3. Verifica che `memcached.service` sia in esecuzione.

```
[ec2-user ~]$ sudo systemctl status memcached.service
# memcached.service - memcached daemon
   Loaded: loaded (/usr/lib/systemd/system/memcached.service; enabled; preset: disabled)
   Active: active (running) since Fri 2025-01-31 22:36:42 UTC; 1s ago
 Main PID: 356294 (memcached)
    Tasks: 10 (limit: 18907)
   Memory: 1.8M
      CPU: 20ms
   CGroup: /system.slice/memcached.service
          ##356294 /usr/bin/memcached -p 11211 -u memcached -m 64 -c 1024 -l 127.0.0.1,::1

Jan 31 22:35:36 ip-1-2-3-4.us-west-2.compute.internal systemd[1]: Started memcached.service - memcached daemon.
```

4. Ora che `memcached` è installato e funzionante, possiamo osservare che funziona inserendo alcuni dati casuali nella cache

`/etc/sysconfig/memcached` Nella `CACHESIZE` variabile è impostato su 64 per impostazione predefinita, ovvero 64 megabyte. Inserendo nella cache una quantità di dati superiore alla dimensione massima consentita, possiamo vedere che la cache `memcached.service` viene riempita e alcuni elementi vengono eliminati utilizzando `memcached-tool` circa 64 MB di memoria.

```
[ec2-user ~]$ for i in $(seq 1 150); do dd if=/dev/random of=$i bs=512k count=1; memcp -s localhost $i; done
[ec2-user ~]$ memcached-tool localhost display
# Item_Size Max_age Pages Count Full? Evicted Evict_Time OOM
2 120B 0s 1 0 no 0 0 0
```

```

39  512.0K      4s    63    126    yes    24     2     0
[ec2-user ~]$ sudo systemctl status memcached.service
# memcached.service - memcached daemon
   Loaded: loaded (/usr/lib/systemd/system/memcached.service; enabled; preset:
disabled)
   Active: active (running) since Fri 2025-01-31 22:36:42 UTC; 7min ago
 Main PID: 356294 (memcached)
    Tasks: 10 (limit: 18907)
   Memory: 66.7M
      CPU: 203ms
   CGroup: /system.slice/memcached.service
           ##356294 /usr/bin/memcached -p 11211 -u memcached -m 64 -c 1024 -l
127.0.0.1,::1

Jan 31 22:36:42 ip-1-2-3-4.us-west-2.compute.internal systemd[1]: Started
memcached.service - memcached daemon.

```

- Utilizzate la `MemoryMax` proprietà per impostare un limite rigido per il `memcached.service` punto in cui, se viene raggiunto, verrà invocato l'OOM Killer. È possibile impostare opzioni aggiuntive per il servizio aggiungendole a un file di override. Questa operazione può essere eseguita modificando direttamente il `/etc/systemd/system/memcached.service.d/override.conf` file o utilizzando in modo interattivo il `edit` comando di `systemctl`

```
[ec2-user ~]$ sudo systemctl edit memcached.service
```

Aggiungi quanto segue all'override per impostare un limite rigido di 32 MB di memoria per il servizio.

```
[Service]
MemoryMax=32M
```

- Dì di `systemd` ricaricarne la configurazione

```
[ec2-user ~]$ sudo systemctl daemon-reload
```

- Osserva che ora `memcached.service` funziona con un limite di memoria di 32 MB.

```
[ec2-user ~]$ sudo systemctl status memcached.service
# memcached.service - memcached daemon

```

```

Loaded: loaded (/usr/lib/systemd/system/memcached.service; enabled; preset:
disabled)
Drop-In: /etc/systemd/system/memcached.service.d
        ##override.conf
Active: active (running) since Fri 2025-01-31 23:09:13 UTC; 49s ago
Main PID: 358423 (memcached)
Tasks: 10 (limit: 18907)
Memory: 1.8M (max: 32.0M available: 30.1M)
CPU: 25ms
CGroup: /system.slice/memcached.service
        ##358423 /usr/bin/memcached -p 11211 -u memcached -m 64 -c 1024 -l
127.0.0.1,::1

Jan 31 23:09:13 ip-1-2-3-4.us-west-2.compute.internal systemd[1]: Started
memcached.service - memcached daemon.

```

8. Il servizio funzionerà normalmente utilizzando meno di 32 MB di memoria, cosa che possiamo verificare caricando meno di 32 MB di dati casuali nella cache e quindi controllando lo stato del servizio.

```
[ec2-user ~]$ for i in $(seq 1 30); do dd if=/dev/random of=$i bs=512k count=1;
memcp -s localhost $i; done
```

```

[ec2-user ~]$ sudo systemctl status memcached.service
# memcached.service - memcached daemon
Loaded: loaded (/usr/lib/systemd/system/memcached.service; enabled; preset:
disabled)
Drop-In: /etc/systemd/system/memcached.service.d
        ##override.conf
Active: active (running) since Fri 2025-01-31 23:14:48 UTC; 3s ago
Main PID: 359492 (memcached)
Tasks: 10 (limit: 18907)
Memory: 18.2M (max: 32.0M available: 13.7M)
CPU: 42ms
CGroup: /system.slice/memcached.service
        ##359492 /usr/bin/memcached -p 11211 -u memcached -m 64 -c 1024 -l
127.0.0.1,::1

Jan 31 23:14:48 ip-1-2-3-4.us-west-2.compute.internal systemd[1]: Started
memcached.service - memcached daemon.

```

9. Ora possiamo memcached utilizzare più di 32 MB di memoria tentando di utilizzare tutti i 64 MB di cache della configurazione predefinita. memcached

```
[ec2-user ~]$ for i in $(seq 1 150); do dd if=/dev/random of=$i bs=512k count=1;
memcp -s localhost $i; done
```

Osserverai che a un certo punto durante il comando precedente ci sono errori di connessione al server. memcached Questo perché OOM Killer ha interrotto il processo a causa della restrizione che gli abbiamo imposto. Il resto del sistema funzionerà normalmente e nessun altro processo verrà preso in considerazione da OOM Killer, in quanto si tratta solo di quelli memcached.service che abbiamo limitato.

```
[ec2-user ~]$ sudo systemctl status memcached.service
# memcached.service - memcached daemon
   Loaded: loaded (/usr/lib/systemd/system/memcached.service; enabled; preset:
disabled)
   Drop-In: /etc/systemd/system/memcached.service.d
           ##override.conf
   Active: failed (Result: oom-kill) since Fri 2025-01-31 23:20:28 UTC; 2s ago
 Duration: 2.901s
   Process: 360130 ExecStart=/usr/bin/memcached -p ${PORT} -u ${USER} -m
${CACHESIZE} -c ${MAXCONN} $OPTIONS (code=killed, signal=KILL)
   Main PID: 360130 (code=killed, signal=KILL)
      CPU: 94ms

Jan 31 23:20:25 ip-1-2-3-4.us-west-2.compute.internal systemd[1]: Started
memcached.service - memcached daemon.
Jan 31 23:20:28 ip-1-2-3-4.us-west-2.compute.internal systemd[1]:
memcached.service: A process of this unit has been killed by the OOM killer.
Jan 31 23:20:28 ip-1-2-3-4.us-west-2.compute.internal systemd[1]:
memcached.service: Main process exited, code=killed, status=9/KILL
Jan 31 23:20:28 ip-1-2-3-4.us-west-2.compute.internal systemd[1]:
memcached.service: Failed with result 'oom-kill'.
```

Limitazione dell'utilizzo delle risorse di processo in AL2 023 utilizzando cgroups

Sebbene se ne consiglia l'uso [Controllo delle risorse con systemd](#), questa sezione tratta l'utilizzo di base delle `libcgroup-tools` utilità di base per limitare l'utilizzo di CPU e memoria dei processi.

Entrambi i metodi sono alternativi all'utilizzo dell'[cpulimit](#) utilità, precedentemente disponibile in [EPEL](#).

L'esempio seguente illustra l'esecuzione dello `stress-ng` stress test (dal `stress-ng` pacchetto) limitando al contempo l'utilizzo della CPU e della memoria utilizzando le utilità del `libcgroup-tools` pacchetto e i regolabili in esso contenuti. `sysfs`

Utilizzalo **`libcgroup-tools`** sulla riga di comando per limitare l'utilizzo delle risorse

1. Installare il pacchetto `libcgroup-tools`.

```
[ec2-user ~]$ sudo dnf install libcgroup-tools
```

2. Crea un comando `cgroup` con `cpu` i controller `memory` e assegnagli un nome (`our-example-limits`). Utilizzando le `-t` opzioni `-a` and per consentire all'`ec2-user` utente di controllare i regolabili del `cgroup`

```
[ec2-user ~]$ sudo cgcreate -a ec2-user -t ec2-user -g memory,cpu:our-example-limits
```

Ora c'è una `/sys/fs/cgroup/our-example-limits/` directory che contiene i file che possono essere usati per controllare ogni tunable.

Note

Amazon Linux 2 utilizza `cgroup-v1` piuttosto di `cgroup-v2` che viene utilizzato su AL2 023. Su AL2, i `sysfs` percorsi sono diversi e ci saranno `/sys/fs/cgroup/memory/our-example-limits` `/sys/fs/cgroup/cpu/our-example-limits` delle cartelle di proprietà `ec2-user` che contengono file che possono essere utilizzati per controllare i limiti di `cgroup`

3. Limita l'utilizzo della memoria di tutti i processi `cgroup` a 100 milioni di byte.

```
[ec2-user ~]$ echo 100000000 > /sys/fs/cgroup/our-example-limits/memory.max
```

Note

Amazon Linux 2 utilizza cgroup-v1 invece di cgroup-v2 quello utilizzato da Amazon Linux 2023. Ciò significa che alcuni regolabili sono diversi. Per limitare l'utilizzo della memoria AL2, viene invece utilizzato il seguente sintonizzabile.

```
[ec2-user ~]$ echo 1000000 > /sys/fs/cgroup/memory/our-example-limits/  
memory.limit_in_bytes
```

4. Limita l'utilizzo della CPU di tutti i processi nel nostro paese cgroup al 10%. Il formato del `cpu.max` file è che `$MAX $PERIOD` limita il gruppo a consumare `$MAX` per tutti `$PERIOD`.

```
[ec2-user ~]$ echo 1000 10000 > /sys/fs/cgroup/our-example-limits/cpu.max
```

Amazon Linux 2 utilizza cgroup-v1 invece di cgroup-v2 quello utilizzato da Amazon Linux 2023. Ciò significa che alcuni regolabili sono diversi, incluso il modo di limitare l'utilizzo della CPU.

5. L'esempio seguente viene eseguito `stress-ng` (che può essere installato eseguendo `dnf install -y stress-ng`) in `our-example-limits` cgroup. Mentre il `stress-ng` comando è in esecuzione, è possibile osservare `top` che è limitato al 10% di CPU tempo.

```
[ec2-user ~]$ sudo cgexec -g memory,cpu:our-example-limits stress-ng --cpu 1
```

6. Pulisci rimuovendo il cgroup

```
[ec2-user ~]$ sudo cgdelete -g memory,cpu:our-example-limits
```

La [documentazione del kernel Linux cgroup-v2](#) contiene dettagli dettagliati sul loro funzionamento. La documentazione relativa alla [CPU](#) e ai controller di [memoria](#) descrive in dettaglio come utilizzare ciascuna opzione ottimizzabile.

Utilizzo AL2023 su AWS

Puoi configurarlo AL2023 per l'uso con altri. Servizi AWS Ad esempio, puoi scegliere un' AL2023 AMI quando avvii un'istanza [Amazon Elastic Compute Cloud](#) (Amazon EC2).

Per queste procedure di configurazione, utilizzi il servizio AWS Identity and Access Management (IAM). Per informazioni complete su IAM, consulta i seguenti materiali di riferimento:

- [AWS Identity and Access Management \(IAM\)](#)
- [Guida per l'utente di IAM](#)

Argomenti

- [Iniziare con AWS](#)
- [AL2023 su Amazon EC2](#)
- [Utilizzo AL2023 in contenitori](#)
- [AL2023 su AWS Elastic Beanstalk](#)
- [Usando AL2 0,23 pollici AWS CloudShell](#)
- [Utilizzo di Amazon ECS basato su AL2 023 AMIs per ospitare carichi di lavoro containerizzati](#)
- [Utilizzo di Amazon Elastic File System su AL2 023](#)
- [Utilizzo di Amazon EMR integrato AL2023](#)
- [Utilizzo AL2023 in AWS Lambda](#)

Iniziare con AWS

Iscriviti per un Account AWS

Se non ne hai uno Account AWS, completa i seguenti passaggi per crearne uno.

Per iscriverti a un Account AWS

1. Apri la <https://portal.aws.amazon.com/billing/registrazione>.
2. Segui le istruzioni online.

Nel corso della procedura di registrazione riceverai una telefonata o un messaggio di testo e ti verrà chiesto di inserire un codice di verifica attraverso la tastiera del telefono.

Quando ti iscrivi a un Account AWS, Utente root dell'account AWS viene creato un. L'utente root dispone dell'accesso a tutte le risorse e tutti i Servizi AWS nell'account. Come best practice di sicurezza, assegna l'accesso amministrativo a un utente e utilizza solo l'utente root per eseguire [attività che richiedono l'accesso di un utente root](#).

AWS ti invia un'email di conferma dopo il completamento della procedura di registrazione. In qualsiasi momento, puoi visualizzare l'attività corrente del tuo account e gestirlo accedendo a <https://aws.amazon.com/> e scegliendo Il mio account.

Crea un utente con accesso amministrativo

Dopo esserti registrato Account AWS, proteggi Utente root dell'account AWS AWS IAM Identity Center, abilita e crea un utente amministrativo in modo da non utilizzare l'utente root per le attività quotidiane.

Proteggi i tuoi Utente root dell'account AWS

1. Accedi [Console di gestione AWS](#) come proprietario dell'account scegliendo Utente root e inserendo il tuo indirizzo Account AWS email. Nella pagina successiva, inserisci la password.

Per informazioni sull'accesso utilizzando un utente root, consulta la pagina [Accedere come utente root](#) nella Guida per l'utente di Accedi ad AWS .

2. Abilita l'autenticazione a più fattori (MFA) per l'utente root.

Per istruzioni, consulta [Abilitare un dispositivo MFA virtuale per l'utente Account AWS root \(console\)](#) nella Guida per l'utente IAM.

Crea un utente con accesso amministrativo

1. Abilita il Centro identità IAM.

Per istruzioni, consulta [Abilitazione del AWS IAM Identity Center](#) nella Guida per l'utente di AWS IAM Identity Center .

2. Nel Centro identità IAM, assegna l'accesso amministrativo a un utente.

Per un tutorial sull'utilizzo di IAM Identity Center directory come fonte di identità, consulta [Configurare l'accesso utente con l'impostazione predefinita IAM Identity Center directory](#) nella Guida per l'AWS IAM Identity Center utente.

Accesso come utente amministratore

- Per accedere come utente del Centro identità IAM, utilizza l'URL di accesso che è stato inviato al tuo indirizzo e-mail quando hai creato l'utente del Centro identità IAM.

Per informazioni sull'accesso utilizzando un utente IAM Identity Center, consulta [AWS Accedere al portale di accesso](#) nella Guida per l'Accedi ad AWS utente.

Assegnazione dell'accesso ad altri utenti

1. Nel Centro identità IAM, crea un set di autorizzazioni conforme alla best practice per l'applicazione di autorizzazioni con il privilegio minimo.

Segui le istruzioni riportate nella pagina [Creazione di un set di autorizzazioni](#) nella Guida per l'utente di AWS IAM Identity Center .

2. Assegna al gruppo prima gli utenti e poi l'accesso con autenticazione unica (Single Sign-On).

Per istruzioni, consulta [Aggiungere gruppi](#) nella Guida per l'utente di AWS IAM Identity Center .

Concessione dell'accesso programmatico

Gli utenti hanno bisogno di un accesso programmatico se vogliono interagire con l' AWS Console di gestione AWS esterno di. Il modo per concedere l'accesso programmatico dipende dal tipo di utente che accede. AWS

Per fornire agli utenti l'accesso programmatico, scegli una delle seguenti opzioni.

Quale utente necessita dell'accesso programmatico?	Per	Come
IAM	(Consigliato) Utilizza le credenziali della console come credenziali temporanee per firmare le richieste programmatiche a,, o. AWS CLI AWS SDKs AWS APIs	Segui le istruzioni per l'interfaccia che desideri utilizzare. <ul style="list-style-type: none"> • Per la AWS CLI, consulta Login for AWS local development nella Guida per l'AWS Command Line Interface utente.

Quale utente necessita dell'accesso programmatico?	Per	Come
		<ul style="list-style-type: none"> Per AWS SDKs, consulta Login for AWS local development nella AWS SDKs and Tools Reference Guide.
Identità della forza lavoro (Utenti gestiti nel centro identità IAM)	Utilizza credenziali temporane e per firmare le richieste programmatiche a AWS CLI, AWS SDKs, o. AWS APIs	<p>Segui le istruzioni per l'interfaccia che desideri utilizzare.</p> <ul style="list-style-type: none"> Per la AWS CLI, vedere Configurazione dell'uso AWS IAM Identity Center nella AWS CLI Guida per l'utente. AWS Command Line Interface Per AWS SDKs gli strumenti e AWS APIs, consulta l'autenticazione di IAM Identity Center nella Guida di riferimento AWS SDKs and Tools.
IAM	Utilizza credenziali temporane e per firmare le richieste programmatiche a AWS CLI, AWS SDKs, o. AWS APIs	Seguendo le istruzioni riportate in Utilizzo delle credenziali temporanee con le AWS risorse nella Guida per l'utente IAM .

Quale utente necessita dell'accesso programmatico?	Per	Come
IAM	(Non consigliato) Utilizza credenziali a lungo termine per firmare richieste programmatiche a AWS CLI,, AWS SDKs o. AWS APIs	<p>Segui le istruzioni per l'interfaccia che desideri utilizzare.</p> <ul style="list-style-type: none"> • Per la AWS CLI, consulta Autenticazione tramite credenziali utente IAM nella Guida per l'utente.AWS Command Line Interface • Per gli strumenti AWS SDKs e gli strumenti, consulta Autenticazione tramite credenziali a lungo termine nella Guida di riferimento agli strumenti e agli AWS SDKs strumenti. • Per AWS APIs, consulta la sezione Gestione delle chiavi di accesso per gli utenti IAM nella Guida per l'utente IAM.

AL2023 su Amazon EC2

Utilizza una delle seguenti procedure per avviare un'istanza Amazon EC2 con un'AMI AL2023 . Puoi scegliere l'AMI standard o l'AMI minima. Per ulteriori informazioni sulle differenze tra AMI standard e AMI minima, consulta [Confronto tra AL2023 standard \(impostazione predefinita\) e minima AMIs](#).

Argomenti

- [Avvio di AL2023 tramite la console Amazon EC2](#)
- [Avvio tramite il parametro SSM e AL2023 AWS CLI](#)
- [Avvio dell' AL2023 AMI più recente utilizzando CloudFormation](#)
- [Avvio AL2023 tramite un ID AMI specifico](#)

- [AL2023 Deprecazione e ciclo di vita degli AMI](#)
- [AL2023 Connessione alle istanze](#)
- [Confronto tra standard e minimi AL2023 AMIs](#)

Avvio di AL2023 tramite la console Amazon EC2

Usa la console Amazon EC2 per avviare un'AMI AL2023 .

Note

Per le istanze basate su ARM, supporta AL2023 solo i tipi di istanza che utilizzano processori Graviton2 o versioni successive. AL2023 non supporta istanze A1.

Attieniti ai passaggi seguenti per avviare un'istanza Amazon EC2 con un'AMI AL2023 dalla console Amazon EC2.

Per avviare un'istanza EC2 con un'AMI AL2023

1. Apri la console Amazon EC2 all'indirizzo <https://console.aws.amazon.com/ec2/>.
2. Nel pannello di navigazione, scegli AMIs.
3. Scegli Immagini pubbliche nel menu a discesa.
4. Inserisci **al2023-ami** nel campo di ricerca.

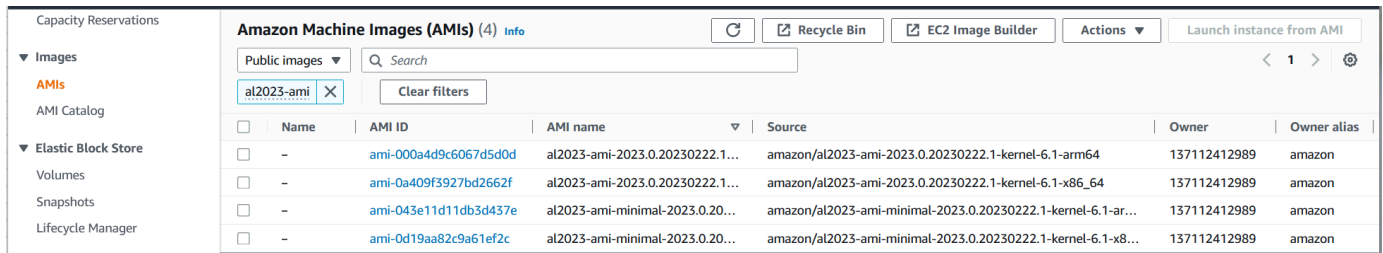
Note

Assicurati che la dicitura amazon compaia nella colonna Alias owner.

5. Seleziona un'immagine dall'elenco. Sotto Origine, puoi determinare se l'AMI è standard o minima. Un nome AL2023 AMI può essere interpretato utilizzando questo formato:

```
'al2023-[ami || ami-minimal]-2023.0.[release build date].[build number]-kernel-[version number]-[arm64 || x86_64]'
```

6. L'immagine seguente mostra un elenco parziale di AL2023 AMIs.



Name	AMI ID	AMI name	Source	Owner	Owner alias
-	ami-000a4d9c6067d5d0d	al2023-ami-2023.0.20230222.1...	amazon/al2023-ami-2023.0.20230222.1-kernel-6.1-arm64	137112412989	amazon
-	ami-0a409f3927bd2662f	al2023-ami-2023.0.20230222.1...	amazon/al2023-ami-2023.0.20230222.1-kernel-6.1-x86_64	137112412989	amazon
-	ami-043e11d11db3d437e	al2023-ami-minimal-2023.0.20...	amazon/al2023-ami-minimal-2023.0.20230222.1-kernel-6.1-ar...	137112412989	amazon
-	ami-0d19aa82c9a61ef2c	al2023-ami-minimal-2023.0.20...	amazon/al2023-ami-minimal-2023.0.20230222.1-kernel-6.1-x8...	137112412989	amazon

Per ulteriori informazioni sul lancio delle istanze Amazon EC2, [consulta la Guida introduttiva alle istanze Amazon EC2 Linux nella Amazon EC2 User Guide](#).

Avvio tramite il parametro SSM e AL2023 AWS CLI

In AWS CLI, è possibile utilizzare il valore del parametro SSM di un'AMI per avviare una nuova istanza di AL2023. Più specificamente, usa uno dei valori dinamici del parametro SSM inclusi nel seguente elenco e aggiungi `/aws/service/ami-amazon-linux-latest/` prima del valore del parametro SSM. Questo ti consente di avviare l'istanza nell'interfaccia AWS CLI.

- `al2023-ami-kernel-default-arm64` per l'architettura `arm64`
- `al2023-ami-minimal-kernel-default-arm64` per l'architettura `arm64` (AMI minima)
- `al2023-ami-kernel-default-x86_64` per l'architettura `x86_64`
- `al2023-ami-minimal-kernel-default-x86_64` per l'architettura `x86_64` (AMI minima)

Note

Ciascuno degli *italic* elementi è un parametro di esempio. Sostituiscili con le tue informazioni.

```
$ aws ec2 run-instances \
  --image-id \
    resolve:ssm:/aws/service/ami-amazon-linux-latest/al2023-ami-kernel-default-x86_64 \
  --instance-type m5.xlarge \
  --region us-east-1 \
  --key-name aws-key-us-east-1 \
  --security-group-ids sg-004a7650
```

Il flag `--image-id` specifica il valore del parametro SSM.

Il flag `--instance-type` specifica il tipo e le dimensioni dell'istanza. Questo flag deve essere compatibile con il tipo di AMI selezionato.

Il `--region` flag specifica il Regione AWS punto in cui crei l'istanza.

Il `--key-name` flag specifica la chiave Regione AWS che viene utilizzata per connettersi all'istanza. Se non fornisci una chiave esistente nella regione in cui crei l'istanza, non puoi connetterti all'istanza tramite SSH.

Il flag `--security-group-ids` specifica il gruppo di sicurezza che determina le autorizzazioni di accesso per il traffico di rete in entrata e in uscita.

Important

È AWS CLI necessario specificare un gruppo di sicurezza esistente che consenta l'accesso all'istanza dal computer remoto tramite portaTCP:22. Senza un gruppo di sicurezza specificato, la nuova istanza viene inserita in un gruppo di sicurezza predefinito. In un gruppo di sicurezza predefinito, l'istanza può connettersi solo alle altre istanze all'interno del tuo VPC.

Per ulteriori informazioni, consulta [Avvio, elencazione e chiusura delle istanze Amazon EC2](#) nella Guida per l'utente di AWS Command Line Interface .

Avvio dell' AL2023 AMI più recente utilizzando CloudFormation

Per avviare un' AL2023 AMI utilizzando CloudFormation, utilizza uno dei seguenti modelli.

Note

Arm64 AMIs Ciascuno richiede tipi di istanza diversi. x86_64 Per ulteriori informazioni consultare [Tipi di istanza Amazon EC2](#)

Modello JSON:

```
{
  "Parameters": {
    "LatestAmiId": {
      "Type": "AWS::SSM::Parameter::Value<AWS::EC2::Image::Id>",
      "Default": "/aws/service/ami-amazon-linux-latest/al2023-ami-minimal-kernel-
default-x86_64"
```

```

    }
  },
  "Resources": {
    "MyEC2Instance": {
      "Type": "AWS::EC2::Instance",
      "Properties": {
        "InstanceType": "t2.large",
        "ImageId": {
          "Ref": "LatestAmiId"
        }
      }
    }
  }
}
}
}
}

```

Modello YAML:

```

Parameters:
  LatestAmiId:
    Type: 'AWS::SSM::Parameter::Value<AWS::EC2::Image::Id>'
    Default: '/aws/service/ami-amazon-linux-latest/al2023-ami-minimal-kernel-default-x86_64'

Resources:
  Instance:
    Type: 'AWS::EC2::Instance'
    Properties:
      InstanceType: 't2.large'
      ImageId: !Ref LatestAmiId

```

Assicurati di sostituire il parametro AMI alla fine della sezione "Predefinito", se necessario. Sono disponibili i seguenti valori di parametri:

- al2023-ami-kernel-6.1-arm64 per l'architettura arm64
- al2023-ami-minimal-kernel-6.1-arm64 per l'architettura arm64 (AMI minima)
- al2023-ami-kernel-6.1-x86_64 per l'architettura x86_64
- al2023-ami-minimal-kernel-6.1-x86_64 per l'architettura x86_64 (AMI minima)

Di seguito sono riportate le specifiche dinamiche del kernel. La versione predefinita del kernel cambia automaticamente con ogni aggiornamento della versione principale del kernel.

- `al2023-ami-kernel-default-arm64` per l'architettura `arm64`
- `al2023-ami-minimal-kernel-default-arm64` per l'architettura `arm64` (AMI minima)
- `al2023-ami-kernel-default-x86_64` per l'architettura `x86_64`
- `al2023-ami-minimal-kernel-default-x86_64` per l'architettura `x86_64` (AMI minima)

Avvio AL2023 tramite un ID AMI specifico

Puoi avviare un'AMI AL2023 specifica utilizzando l'ID AMI. Puoi determinare l'ID AMI AL2023 necessaria consultando l'elenco delle AMI nella console Amazon EC2. Oppure puoi usare AWS Systems Manager. Se usi Systems Manager, assicurati di selezionare l'alias AMI tra quelli elencati nella sezione precedente. Per ulteriori informazioni, consulta [Query per l'ultima AMI Amazon Linux IDs con AWS Systems Manager Parameter Store](#).

AL2023 Deprecazione e ciclo di vita degli AMI

Ogni nuova AL2023 versione include una nuova AMI. Al momento della registrazione, l'AMI viene contrassegnata con una data di obsolescenza. La data di obsolescenza per ogni AL2023 AMI è di 90 giorni dal momento in cui è stata rilasciata, in modo da corrispondere al periodo di tempo offerto per ogni singola versione del kernel. [Kernel Live Patching attivo AL2023](#)

Note

La data di obsolescenza di 90 giorni si riferisce a una singola AMI e non al periodo di assistenza AL2023 [Cadenza di rilascio](#) o al periodo di supporto del prodotto.

Per ulteriori informazioni sulla deprecazione degli AMI, consulta [Deprecare un AMI](#) nella Amazon EC2 User Guide.

L'uso regolare di un'AMI aggiornata per avviare un'istanza garantisce che quest'ultima venga avviata con i più recenti aggiornamenti di sicurezza, incluso un kernel aggiornato. Se avvii una versione precedente di un'AMI e poi applichi gli aggiornamenti, esiste un periodo di tempo per cui l'istanza non dispone dei più recenti aggiornamenti di sicurezza. Per assicurarti di utilizzare l'AMI più recente, ti consigliamo di usare i parametri SSM.

Per ulteriori informazioni sull'uso dei parametri SSM per avviare un'istanza, consulta:

- [Avvio tramite il parametro SSM e AL2023 AWS CLI](#)

- [Avvio dell' AL2023 AMI più recente utilizzando CloudFormation](#)

AL2023 Connessione alle istanze

Usa SSH o AWS Systems Manager per connetterti alla tua istanza. AL2023

Connettersi all'istanza tramite SSH

Per istruzioni su come usare SSH per connettersi a un'istanza, consulta [Connettiti alla tua istanza Linux usando SSH](#) nella Amazon EC2 User Guide.

Connect alla tua istanza utilizzando AWS Systems Manager

Per istruzioni su come utilizzare la connessione AWS Systems Manager a un' AL2023 istanza, consulta [Connettiti alla tua istanza Linux utilizzando Session Manager](#) nella Amazon EC2 User Guide.

Utilizzo di Amazon EC2 Instance Connect

L'AMI AL2023, esclusa l'AMI minima, viene fornita con l'agente EC2 Instance Connect installato per impostazione predefinita. Per utilizzare EC2 Instance Connect con un' AL2023 istanza lanciata dall'AMI minima, devi installare il `ec2-instance-connect` pacchetto. Per istruzioni sull'uso di EC2 Instance Connect, consulta [Connettiti alla tua istanza Linux con EC2 Instance Connect](#) nella Amazon EC2 User Guide.

Confronto tra standard e minimi AL2023 AMIs

Puoi avviare un'istanza Amazon EC2 con un'AMI AL2023 standard (predefinita) o minima. Per istruzioni su come avviare un'istanza Amazon EC2 con il tipo di AMI standard o minimo, consulta [AL2023 su Amazon EC2](#)

L' AL2023 AMI standard viene fornita con tutte le applicazioni e gli strumenti più comunemente utilizzati installati. Consigliamo di usare l'AMI standard se desideri iniziare rapidamente e non ti interessa personalizzare l'AMI.

L' AL2023 AMI minima è la versione semplificata di base che contiene solo gli strumenti e le utilità di base necessari per eseguire il sistema operativo (OS). Consigliamo di usare l'AMI minima se desideri avere il minor ingombro possibile del sistema operativo. L'AMI minima offre un utilizzo leggermente ridotto dello spazio su disco e una migliore efficienza in termini di costi a lungo termine. L'AMI minima è adatta se desideri un sistema operativo più piccolo e hai problemi a installare manualmente strumenti e applicazioni.

L'immagine del contenitore è più vicina all'AMI AL2023 minima nel set di pacchetti.

Confronto dei pacchetti installati sulle immagini Amazon Linux 2023

Un confronto dei RPMs presente sulle immagini AL2023 AMI, Minimal AMI e Container.

Pacchetto	AMI	AMI minima	Contenitore
acl	2.3.1		
acpid	2.0.32		
alternatives	1.15	1.15	1.15
amazon-chrony-config	4.3	4.3	
amazon-ec2-net-utils	2.5.1	2.5.1	
amazon-linux-repo-cdn			2023,620241031
amazon-linux-repo-s3	2023,620241031	2023,620241031	
amazon-linux-sb-keys	2023,1	2023,1	
amazon-rpm-config	228		
amazon-ssm-agent	3,3,987,0		
amd-ucode-firmware	20210208 (marzo)	20210208 (novembre)	
at	3.1.23		
attr	2.5.1		

Pacchetto	AMI	AMI minima	Contenitore
audit	30,6	3.0.6	
audit-libs	3.0.6	3.0.6	3.0.6
aws-cfn-bootstrap	2.0		
awscli-2	2,15,30	2,15,30	
basesystem	11	11	11
bash	5,2,15	5,2,15	5,2,15
bash-completion	2.11		
bc	1,07,1		
bind-libs	9,18,28		
bind-license	9,18,28		
bind-utils	9,18,28		
binutils	2,39		
boost-filesystem	1,75,0		
boost-system	1,75,0		
boost-thread	1,75,0		
bzip2	1.0.8		
bzip2-libs	1.0.8	1.0.8	1.0.8
ca-certificates	2023,2,68	2023,2,68	2023,2,68
c-ares	1.19.1		

Pacchetto	AMI	AMI minima	Contenitore
checkpolicy	3.4	3.4	
chkconfig	1.15		
chrony	4.3	4.3	
cloud-init	222,2	222,2	
cloud-init-cfg-ec2	222,2	222,2	
cloud-utils-growpart	0,31	0,31	
coreutils	8,32	8,32	
coreutils-common	8,32	8,32	
coreutils-single			8,32
cpio	2,13	2,13	
cracklib	2,9,6	29.6	
cracklib-dicts	29.6	29.6	
crontabs	1.11		
crypto-policies	20220428	20220428	20220428
crypto-policies-scripts	20220428		
cryptsetup	2.6.1		
cryptsetup-libs	2.6.1	2.6.1	

Pacchetto	AMI	AMI minima	Contenitore
curl-minimal	8,5,0	8,5,0	8,5,0
cyrus-sasl-lib	2,1,27	2,1,27	
cyrus-sasl-plain	2,1,27		
dbus	1,12,28	1,12,28	
dbus-broker	32	32	
dbus-common	1,12,28	1,12,28	
dbus-libs	1,12,28	1,12,28	
device-mapper	1,02,185	1,02,185	
device-mapper-libs	1,02,185	1,02,185	
diffutils	3.8	3.8	
dnf	4.14.0	4.14.0	4.14.0
dnf-data	4.14.0	4.14.0	4.14.0
dnf-plugin-n-release-notification	1.2	1.2	
dnf-plugins-core	4.3.0	4.3.0	
dnf-plugin-support-info	1.2	1.2	
dnf-utils	4.3.0		
dosfstools	4.2		

Pacchetto	AMI	AMI minima	Contenitore
dracut	055	055	
dracut-config-ec2	3.0	3.0	
dracut-config-generic	055	055	
dwz	0,14		
dyninst	102,1		
e2fsprogs	1,446,5	1,46,5	
e2fsprogs-libs	1,46,5	1,46,5	
ec2-hibinit-agent	1.0.8		
ec2-instance-connect	1.1		
ec2-instance-connect-selinux	1.1		
ec2-utils	2.2.0	2.2.0	
ed	1.14.2		
efi-filesystem	5	5	
efi-srpm-macros	5		
efivar	38	38	
efivar-libs	38	38	

Pacchetto	AMI	AMI minima	Contenitore
elfutils-debuginfod-client	0.188		
elfutils-default-yama-scope	0.188	0.188	0.188
elfutils-libelf	0.188	0.188	0.188
elfutils-libs	0.188	0.188	0.188
ethtool	5,15		
expat	2.5.0	2.5.0	2.5.0
file	5,39	5,39	
file-libs	5,39	5,39	5,39
filesystem	3,14	3,14	3,14
findutils	4.8.0	4.8.0	
fonts-srpm-macros	2.0.5		
fstrm	0.6.1		
fuse-libs	2,9,9	29.9	
gawk	5.1.0	5.1.0	5.1.0
gdbm-libs	1,19	1,19	1,19
gdisk	1.0.8	1.0.8	
gettext	0,21	0,21	

Pacchetto	AMI	AMI minima	Contenitore
gettext-libs	0,21	0,21	
ghc-srpm-macros	1.5.0		
glib2	2,74,7	2,74,7	2,74,7
glibc	2,34	2,34	2,34
glibc-all-langpacks	2,34	2,34	
glibc-common	2,34	2,34	2,34
glibc-gconv-extra	2,34		
glibc-locale-source	2,34	2,34	
glibc-minimal-langpack			2,34
gmp	62,1	6.2.1	6.2.1
gnupg2-minimal	2.3.7	2.3.7	2.3.7
gnutls	3.8.0	3.8.0	
go-srpm-macros	3.2.0		
gpgme	1.15.1	1.15.1	1.15.1
gpm-libs	1,20.7		
grep	3.8	3.8	3.8
groff-base	1,22,4	1,22,4	
grub2-common	2,06	2,06	

Pacchetto	AMI	AMI minima	Contenitore
grub2-efi-aa64-ec2	2,06 (aarch64)	2.06 (aarch64)	
grub2-efi-x64-ec2	2,06 (x86_64)	2,06 (x86_64)	
grub2-pc-modules	2,06	2,06	
grub2-tools	2,06	2,06	
grub2-tools-minimal	2,06	2,06	
grubby	8,40	8,40	
gssproxy	0.8.4		
gzip	1.12	1.12	
hostname	3,23	3,23	
hunspell	1.7.0		
hunspell-en	0,20140811,1		
hunspell-en-GB	0,20140811,1		
hunspell-en-US	0,20140811,1		
hunspell-filesystem	1.7.0		
hwdata	0,384	0,384	
info	6.7		
inih	49	49	

Pacchetto	AMI	AMI minima	Contenitore
initscripts	10,09	10,09	
iproute	6.10.0	6.10.0	
iputils	20210202	20210202	
irqbalance	1.9.0	1.9.0	
jansson	2.14	2.14	
jemalloc	52,1		
jitterentropy	3.4.1	3.4.1	
jq	1.7.1	1.7.1	
json-c	0,14	0,14	0,14
kbd	2.4.0	2.4.0	
kbd-misc	2.4.0	2.4.0	
kernel	6,1112	6,1112	
kernel-libbpf	6,1112	6,1112	
kernel-li vepatch-repo- s3	2023,620241031	2023,620241031	
kernel-srpm- macros	1.0		
kernel-tools	6,1112		
keyutils	1.6.3		
keyutils-libs	1.6.3	1.6.3	1.6.3

Pacchetto	AMI	AMI minima	Contenitore
kmod	29	29	
kmod-libs	29	29	
kpatch-runtime	0,9,7		
krb5-libs	1,21,3	1,21,3	1,21,3
less	608	608	
libacl	2.3.1	2.3.1	2.3.1
libaio	0,3111		
libarchive	3,7,4	3,7,4	3,7,4
libargon2	20171227	20171227	
libassuan	2,5,5	2,5,5	2,5,5
libattr	2.5.1	2.5.1	2.5.1
libbasicobjects	0,11		
libblkid	2,37,4	2,37,4	2,37,4
libcap	2,48	2,48	2,48
libcap-ng	0.8.2	0.8.2	0.8.2
libcbor	0.7.0	0.7.0	
libcollection	0.7.0		
libcom_err	1,446,5	1,46,5	1,46,5
libcomps	01,20	01,20	01,20
libconfig	17.2		

Pacchetto	AMI	AMI minima	Contenitore
libcurl-minimal	8,5,0	8,5,0	8,5,0
libdb	5,3,28	5,3,28	
libdhash	0,50		
libdnf	0,69,0	0,69,0	0,69,0
libeconf	0,40	0,40	
libedit	3.1	3.1	
libev	4,33		
libevent	2,1,12		
libfdisk	2,37,4	2,37,4	
libffi	34.4	34.4	34.4
libfido2	1.10.0	1.10.0	
libgcc	11.4.1	11.4.1	11.4.1
libgcrypt	1.10.2	1.10.2	1.10.2
libgomp	11.4.1	11.4.1	11.4.1
libgpg-error	1,42	1,42	1,42
libibverbs	48,0		
libidn2	2.3.2	2.3.2	2.3.2
libini_config	1.3.1		
libkcapi	1.4.0	1.4.0	
libkcapi-hmacalc	1.4.0	1.4.0	

Pacchetto	AMI	AMI minima	Contenitore
libldb	26.2		
libmaxminddb	1.5.2		
libmetalink	0,13		
libmnl	1.0.4	1.0.4	
libmodulemd	2.13.0	2.13.0	2.13.0
libmount	2,37,4	2,37,4	2,37,4
libnfsidmap	2,5,4		
libnghttp2	1,59,0	1,59,0	1,59,0
libnl3	3.5.0		
libpath_utils	0,2,1		
libpcap	1.10.1		
libpipeline	1.5.3	1.5.3	
libpkgconf	1.8.0		
libpsl	0,21,1	0,21,1	0,21,1
libpwquality	1.4.4	1.4.4	
libref_array	0,1,5		
librepo	1,14,5	1,14,5	1,14,5
libreport-filesystem	2,15,2	2,15,2	2,15,2
libseccomp	2.5.3	2.5.3	
libselinux	3.4	3.4	3.4

Pacchetto	AMI	AMI minima	Contenitore
libselinux- utils	3.4	3.4	
libsemanage	3.4	3.4	
libsepol	3.4	3.4	3.4
libsigsegv	2,13	2,13	2,13
libsmartcols	2,37,4	2,37,4	2,37,4
libsolv	0,7,22	0,7,22	0,7,22
libss	1,446,5	1,46,5	
libsss_certmap	2,9,4		
libsss_idmap	2,9,4		
libsss_ns s_idmap	2,9,4		
libsss_sudo	2,9,4		
libstdc++	11.4.1	11.4.1	11.4.1
libstoragemgmt	1.9.4		
libtalloc	2.3.4		
libtasn1	4,19,0	4,19,0	4,19,0
libtdb	1.4.7		
libtevent	0.13.0		
libtextstyle	0,21	0,21	
libtirpc	1.3.3		

Pacchetto	AMI	AMI minima	Contenitore
libunistring	0,9,10	0,9,10	0,9,10
libuser	0,63	0,63	
libutempter	1.2.1	1.2.1	
libuuid	2,37,4	2,37,4	2,37,4
libuv	1.47.0		
libverto	0,32	0,32	0,32
libverto-libev	0,32		
libxcrypt	4,4,33	4,4,33	4,4,33
libxml2	2,10,4	210.4	210.4
libyaml	02,5	02,5	02,5
libzstd	1,5,5	1,5,5	1,5,5
linux-firmware-whence	20210208 (novembre)	20210208 (novembre)	
lm_sensors-libs	3.6.0		
lmdb-libs	0,9,29		
logrotate	3,20,1	3,20,1	
lsof	4,94,0		
lua-libs	54.4	5.4.4	5.4.4
lua-srpm-macros	1		
lz4-libs	1.9.4	1.9.4	1.9.4
man-db	2,9,3	29.3	

Pacchetto	AMI	AMI minima	Contenitore
man-pages	5,10		
microcode_ctl	2.1 (x86_64)	2.1 (x86_64)	
mpfr	4.1.0	4.1.0	4.1.0
nano	5.8		
ncurses	6.2	6.2	
ncurses-base	6.2	6.2	6.2
ncurses-libs	6.2	6.2	6.2
nettle	3.8	3.8	
net-tools	2.0	2.0	
newt	0,52,21		
nfs-utils	2,5,4		
npth	1.6	1.6	1.6
nspr	4,35,0		
nss	3,90,0		
nss-softokn	3,90,0		
nss-softokn-freebl	3,90,0		
nss-sysinit	3,90,0		
nss-util	3,90,0		
ntsysv	1.15		
numactl-libs	2.0,14	2.0,14	

Pacchetto	AMI	AMI minima	Contenitore
ocaml-srpm-macros	6		
oniguruma	6,9,7,1	6,9,7,1	
openblas-srpm-macros	2		
openldap	2,4,57	2,4,57	
openssh	8.7p1	8,7p1	
openssh-clients	8,7p1	8,7p1	
openssh-server	8,7p1	8,7p1	
openssl	3,0,8	30,8	
openssl-lib	30,8	30,8	30,8
openssl-pkcs11	0,4,12	0,4,12	
os-prober	1,77	1,77	
p11-kit	0,24,1	0,24,1	0,24,1
p11-kit-trust	0,24,1	0,24,1	0,24,1
package-notes-srpm-macros	0.4		
pam	1.5.1	1.5.1	
parted	3.4		
passwd	0,80	0,80	
pciutils	3.7.0	3.7.0	

Pacchetto	AMI	AMI minima	Contenitore
pciutils-libs	3.7.0	3.7.0	
pcre2	10,40	10,40	10,40
pcre2-syntax	10,40	10,40	10,40
perl-Carp	1,50		
perl-Class-Struct	0,66		
perl-constant	1,33		
perl-DynaLoader	1,47		
perl-Encode	3,15		
perl-Errno	1,30		
perl-Exporter	5,74		
perl-Fcntl	1.13		
perl-File-Basename	2,85		
perl-File-Path	2,18		
perl-File-stat	1,09		
perl-File-Temp	0,231,100		
perl-Getopt-Long	2,52		
perl-Getopt-Std	1.12		
perl-HTTP-Tiny	0,078		

Pacchetto	AMI	AMI minima	Contenitore
perl-if	0,60,800		
perl-interpret	5,32,1		
perl-IO	1,43		
perl-IPC-Open3	1,21		
perl-libs	5,32,1		
perl-MIME-Base64	3,16		
perl-mro	1,23		
perl-overload	1,31		
perl-overloading	0,02		
perl-parent	0,238		
perl-PathTools	3,78		
perl-Pod-Escapes	1,07		
perl-podlators	4,14		
perl-Pod-Perldoc	3,28,01		
perl-Pod-Simple	3,42		
perl-Pod-Usage	2,01		
perl-POSIX	1,94		

Pacchetto	AMI	AMI minima	Contenitore
perl-Scalar-List-Utils	1,56		
perl-SelectSaver	1.02		
perl-Socket	2,032		
perl-srpm-macros	1		
perl-Storable	3,21		
perl-subst	1,03		
perl-Symbol	1,08		
perl-Term-ANSIColor	5,01		
perl-Term-Cap	1,17		
perl-Text-ParseWords	3,30		
perl-Text-Tabs+Wrap	2021,0726		
perl-Time-Local	1,300		
perl-vars	1,05		
pkgconf	1.8.0		
pkgconf-m4	1.8.0		
pkgconf-pkg-config	1.8.0		

Pacchetto	AMI	AMI minima	Contenitore
policycoreutils	3.4	3.4	
policycoreutils-python-utils	3.4		
popt	1,18	1,18	1,18
procps-ng	3,3,17	3,3,17	
protobuf-c	1.4.1		
psacct	6,6,4		
psmisc	23,4	23,4	
publicsuffix-list-dafsa	20240212	20240212	20240212
python3	3,9,16	3,9,16	3,9,16
python3-attrs	20,3,0	20,3,0	
python3-audit	30.6	3.0.6	
python3-awscli	0,19,19	0,19,19	
python3-babel	2,9,1	29.1	
python3-cffi	1,14,5	1,14,5	
python3-chardet	4.0.0	4.0.0	
python3-colorama	04.4	04.4	
python3-configobj	50.6	5.0.6	

Pacchetto	AMI	AMI minima	Contenitore
python3-cryptography	36,0	36,0	
python3-daemon	2.3.0		
python3-dateutil	28.1	28.1	
python3-dbus	1,2,18	1,2,18	
python3-distro	1.5.0	1.5.0	
python3-dnf	4.14.0	4.14.0	4.14.0
python3-dnf-plugins-core	4.3.0	4.3.0	
python3-docutils	0,16	0,16	
python3-gpg	1.15.1	1.15.1	1.15.1
python3-hawkey	0,69,0	0,69,0	0,69,0
python3-idna	(2.10)	(2.10)	
python3-jinja2	211,3	2.11.3	
python3-jmespath	0.10.0	0.10.0	
python3-jsonpatch	1,21	1,21	
python3-jsonpointer	2.0	2.0	

Pacchetto	AMI	AMI minima	Contenitore
python3-j sonschema	3.2.0	3.2.0	
python3-l ibcomps	01,20	01,20	01,20
python3-libdnf	0,69,0	0,69,0	0,69,0
python3-libs	3,9,16	3,9,16	3,9,16
python3-l ibselinux	3.4	3.4	
python3-l ibsemanage	3.4	3.4	
python3-l ibstoragemgmt	1.9.4		
python3-l ockfile	0.12.2		
python3-m arkupsafe	1.1.1	1.1.1	
python3-n etifaces	0,10,6	0,10,6	
python3-o authlib	3.0.2	3.0.2	
python3-pip- wheel	21,31	21,31	21,31
python3-ply	3,11	3,11	
python3-p olicycoreutils	3.4	3.4	

Pacchetto	AMI	AMI minima	Contenitore
python3-p rettytable	0.7.2	0.7.2	
python3-prompt- toolkit	3,0,24	3,0,24	
python3-p ycparser	2,20	2,20	
python3-p yrsistent	0,17,3	0,17,3	
python3-p yserial	3.4	3.4	
python3-pysocks	1.7.1	1.7.1	
python3-pytz	2022,7,1	2022,7,1	
python3-pyyaml	5.4.1	5.4.1	
python3-r equests	2,25,1	2,25,1	
python3-rpm	4,161,3	4,161,3	4,161,3
python3-ruamel- yaml	0,16,6	0,16,6	
python3-ruamel- yaml-clib	0,12	0,1,2	
python3-setools	4.4.1	4.4.1	
python3-s etuptools	59,60	59,6,0	

Pacchetto	AMI	AMI minima	Contenitore
python3-s etuptools- wheel	59,6,0	59,6,0	59,6,0
python3-six	1.15.0	1.15.0	
python3-systemd	235	235	
python3-urllib3	1,25,10	1,25,10	
python3-wcwidth	0,2,5	02,5	
python-chevron	0.13.1		
python-srpm- macros	3.9		
quota	4,06		
quota-nls	4,06		
readline	8.1	8.1	8.1
rng-tools	6,14	6,14	
rootfiles	8.1	8.1	
rpcbind	1.2.6		
rpm	4,161,3	4,161,3	4,161,3
rpm-build-libs	4,161,3	4,161,3	4,161,3
rpm-libs	4,161,3	4,161,3	4,161,3
rpm-plugin- selinux	4,161,3	4,161,3	

Pacchetto	AMI	AMI minima	Contenitore
rpm-plugin-systemd-inhibit	4,161,3	4,161,3	
rpm-sign-libs	4,161,3	4,161,3	4,161,3
rsync	3.2.6		
rust-srpm-macros	21		
sbsigntools	0.9.4	0.9.4	
screen	4.8.0		
sed	4.8	4.8	4.8
selinux-policy	381,45	381,45	
selinux-policy-targeted	381,45	381,45	
setup	2,13,7	2,13,7	2,13,7
shadow-utils	4.9	4.9	
slang	2.3.2		
sqlite-libs	3,4,0	3,4,0	3,4,0
sssd-client	2,9,4		
sssd-common	2,9,4		
sssd-kcm	2,9,4		
sssd-nfs-idmap	2,9,4		
strace	6.8		

Pacchetto	AMI	AMI minima	Contenitore
sudo	1,9,15	1,9,15	
sysctl-defaults	1.0	1.0	
sysstat	12,5,6		
systemd	252,23	252,23	
systemd-libs	252,23	252,23	
systemd-n networkd	252,23	252,23	
systemd-pam	252,23	252,23	
systemd-r esolved	252,23	252,23	
systemd-udev	252,23	252,23	
system-release	2023,620241031	2023,620241031	2023,620241031
systemtap- runtime	4,8		
tar	1,34	1,34	
tbb	2020,3		
tcpdump	4,99,1		
tcsh	6,24,07		
time	1.9		
traceroute	2.1.3		
tzdata	2024a	2024a	2024a

Pacchetto	AMI	AMI minima	Contenitore
unzip	6.0		
update-motd	2.2	2.2	
userspace-rcu	0.12.1	0.12.1	
util-linux	2,37,4	2,37,4	
util-linux-core	2,37,4	2,37,4	
vim-common	9,0,2153		
vim-data	9,0,2153	9,0,2153	
vim-enhanced	9,0,2153		
vim-filesystem	9,0,2153		
vim-minimal	9,0,2153	9,0,2153	
wget	1,21,3		
which	2,21	2,21	
words	3.0		
xfsdump	3,1,11		
xfsplogs	5,18,0	5,18,0	
xxd	9,0,2153		
xxhash-libs	0.8.0		
xz	52,5	5.2.5	
xz-libs	5.2.5	5.2.5	5.2.5
yum	4.14.0	4.14.0	4.14.0

Pacchetto	AMI	AMI minima	Contenitore
zip	3.0		
zlib	1,2,11	1,2,11	1,2,11
zram-generator	1.1.2	1.1.2	
zram-generator-defaults	1.1.2	1.1.2	
zstd	1,5,5	1,5,5	

Utilizzo AL2023 in contenitori

Note

Per ulteriori informazioni su come utilizzare per ospitare carichi AL2023 di lavoro containerizzati su Amazon ECS, consulta [AL2023 per gli host di container Amazon ECS](#)

Esistono diversi modi che AL2023 possono essere utilizzati all'interno dei contenitori a seconda del caso d'uso. [AL2023 Immagine del contenitore di base](#) È molto simile a un'immagine di contenitore Amazon Linux 2 e all'AMI AL2023 minima.

Per gli utenti esperti, offriamo un'immagine minimale del contenitore, introdotta nella release AL2023 .2, insieme alla documentazione che descrive come creare contenitori [bare-bone](#).

AL2023 può essere utilizzato anche per ospitare carichi di lavoro containerizzati, basati su immagini di container o contenitori AL2023 basati su altre distribuzioni Linux. Puoi usare [AL2023 per gli host di container Amazon ECS](#) oppure usare direttamente i pacchetti di runtime dei container forniti. I `nerdctl` pacchetti `docker`, `containerd`, e sono disponibili per essere installati e utilizzati. AL2023

Argomenti

- [Utilizzo dell'immagine del contenitore di AL2023 base](#)
- [AL2023 Immagine minima del contenitore](#)
- [Creazione di immagini di contenitori essenziali AL2023](#)

- [Confronto dei pacchetti installati sulle immagini di container Amazon Linux 2023](#)
- [Confronto dei pacchetti installati sulle immagini di container e AMI minime Amazon Linux 2023](#)

Utilizzo dell'immagine del contenitore di AL2023 base

L'immagine del AL2023 contenitore è costruita con gli stessi componenti software inclusi nell' AL2023 AMI. È disponibile per l'uso in qualsiasi ambiente come immagine di base per i carichi di lavoro di Docker. Se usi già l'AMI Amazon Linux per le applicazioni in [Amazon Elastic Compute Cloud](#) (Amazon EC2), puoi containerizzare le tue applicazioni con l'immagine di container Amazon Linux.

Usa l'immagine del contenitore Amazon Linux nel tuo ambiente di sviluppo locale e poi invia l'applicazione all' AWS utilizzando di [Amazon Elastic Container Service](#) (Amazon ECS). Per ulteriori informazioni, consulta la pagina dedicata all'[utilizzo di immagini Amazon ECR con Amazon ECS](#) nella Guida per l'utente di Amazon Elastic Container Registry.

L'immagine di container Amazon Linux è disponibile su Amazon ECR Public. Puoi fornire un feedback AL2023 tramite il tuo AWS rappresentante designato o segnalando un problema nel repository [amazon-linux-2023](#) su. GitHub

Per estrarre l'immagine di container Amazon Linux da Amazon ECR Public

1. Autentica il client Docker nel registro Amazon Linux Public. I token di autenticazione sono validi 12 ore. Per ulteriori informazioni, consulta [Autenticazione del registro privato](#) nella Guida per l'utente di Amazon Elastic Container Registry.

Note

Il `get-login-password` comando è supportato utilizzando l'ultima versione della versione 2. AWS CLI Per ulteriori informazioni, consulta [Installazione dell' AWS Command Line Interface](#) nella Guida per l'utente dell'AWS Command Line Interface .

```
$ aws ecr-public get-login-password --region us-east-1 | docker login --username  
AWS --password-stdin public.ecr.aws
```

L'output è il seguente.

```
Login succeeded
```

2. Estrai l'immagine di container Amazon Linux eseguendo il comando `docker pull`. Per visualizzare l'immagine del container Amazon Linux nella galleria di Amazon ECR Public, consulta la [galleria di Amazon ECR Public - amazonlinux](#).

Note

Quando si estrae l'immagine del AL2023 Docker contenitore, è possibile utilizzare i tag in uno dei seguenti formati:

- Per ottenere la versione più recente dell'immagine del AL2023 contenitore, usa il `:2023` tag.
- Per ottenere una versione specifica di AL2023, puoi utilizzare il seguente formato:
 - `:2023.[0-7 release quarter].[release date].[build number]`

Gli esempi seguenti utilizzano il tag `:2023` e recuperano l'immagine del contenitore più recente disponibile di AL2023.

```
$ docker pull public.ecr.aws/amazonlinux/amazonlinux:2023
```

3. (Opzionale) Esegui il container a livello locale.

```
$ docker run -it --security-opt seccomp=unconfined public.ecr.aws/amazonlinux/amazonlinux:2023 /bin/bash
```

Per estrarre l'immagine del AL2023 contenitore da Docker Hub

1. Estrai l'immagine del AL2023 contenitore usando il `docker pull` comando.

```
$ docker pull amazonlinux:2023
```

2. (Opzionale) Esegui il container a livello locale.

```
$ docker run -it amazonlinux:2023 /bin/bash
```

Note

L'immagine del contenitore di AL2023 utilizza solo il gestore di `dnf` pacchetti per installare i pacchetti software. Ciò significa che non esiste alcun comando `amazon-linux-extras` né uno equivalente da usare per software aggiuntivi.

AL2023 Immagine minima del contenitore

Note

Le immagini standard del AL2023 contenitore sono adatte alla maggior parte dei casi d'uso e l'adattamento all'immagine minima del contenitore è probabilmente più faticoso che adattarsi all'immagine del contenitore di AL2023 base.

L'immagine AL2023 minima del contenitore, introdotta in AL2023 .2, differisce dall'immagine del contenitore di base perché contiene solo i pacchetti minimi necessari per installare altri pacchetti. L'immagine minima del contenitore è progettata per essere un insieme minimo di pacchetti, non un comodo set di pacchetti.

L'immagine AL2023 minima del contenitore è creata a partire da componenti software già disponibili in AL2023. La differenza principale nell'immagine minimale del contenitore consiste nell'utilizzo `microdnf` del gestore dei `dnf` pacchetti piuttosto che in quello Python basato su tutte le funzionalità `dnf`. Ciò consente di ridurre al minimo l'immagine del contenitore, con il vantaggio di non disporre del set completo di funzionalità del gestore di `dnf` pacchetti, incluso nell'immagine del AL2023 AMIs contenitore di base.

L'immagine AL2023 minimale del contenitore costituisce la base dell'ambiente di runtime `provided.al2023` AWS Lambda.

Per un elenco dettagliato dei pacchetti inclusi nell'immagine minima del contenitore, consulta [Confronto dei pacchetti installati sulle immagini di container Amazon Linux 2023](#).

Dimensioni dell'immagine di container minima

Poiché l'immagine AL2023 minima del contenitore contiene meno pacchetti rispetto all'immagine del contenitore di AL2023 base, è anche notevolmente più piccola. La tabella seguente confronta le opzioni relative all'immagine del contenitore delle versioni correnti e precedenti di Amazon Linux.

Note

Le dimensioni delle immagini sono quelle mostrate nella pagina della [Galleria pubblica di Amazon ECR dedicata ad Amazon Linux](#).

Immagine	Versione	Dimensioni dell'immagine	Nota
Amazon Linux (1AL1)	2018/03/0,20230918,0	62,3 MB	Solo x86-64
Amazon Linux 2	2,0,20230926,0	64,2 MB	aaarch64 è di 1,6 MB più grande di x86-64
Immagine di container di base Amazon Linux 2023	20232,20231002,0	52,4 MB	
Immagine di container minima Amazon Linux 2023	2023.2.20231002.0-minimal	35,2 MB	

Utilizzo dell'immagine Minimal Container AL2023

L'immagine AL2023 minima del contenitore è disponibile su ECR e il `2023-minimal` tag punterà sempre all'immagine del contenitore minimo AL2023 basata più recente, mentre il `minimal` tag può essere aggiornato a una versione più recente di Amazon Linux rispetto AL2023 a.

Puoi estrarre questi tag usando `docker` il seguente esempio:

```
$ docker pull public.ecr.aws/amazonlinux/amazonlinux:minimal
```

```
$ docker pull public.ecr.aws/amazonlinux/amazonlinux:2023-minimal
```

L'esempio seguente mostra un file Dockerfile che prende l'immagine minima del contenitore e installa GCC su di essa:

```
FROM public.ecr.aws/amazonlinux/amazonlinux:2023-minimal
RUN dnf install -y gcc && dnf clean all
```

Creazione di immagini di contenitori essenziali AL2023

L'immagine del contenitore AL2023 è costruita con gli stessi componenti software inclusi nell'AMI AL2023. Include un software che consente al livello di contenitore di base di comportarsi in modo simile all'esecuzione su un'istanza Amazon EC2, ad esempio il gestore di pacchetti. dnf Questa sezione spiega come costruire da zero un contenitore che includa solo le dipendenze minime necessarie per un'applicazione.

Note

Le immagini standard dei AL2023 contenitori sono adatte alla maggior parte dei casi d'uso. L'uso dell'immagine di container standard semplifica la creazione sulla base dell'immagine. Un'immagine contenitore semplice rende più difficile la creazione sulla base dell'immagine.

Per creare un container con dipendenze minime essenziali per un'applicazione

1. Determina le dipendenze di runtime. Questo aspetto varia a seconda dell'applicazione.
2. Crea un Dockerfile/Containerfile che consenta di sviluppare FROM scratch. Il seguente esempio di Dockerfile può essere usato per creare un container che include solo la shell bash e le relative dipendenze.

```
FROM public.ecr.aws/amazonlinux/amazonlinux:2023 as build
RUN mkdir /sysroot
RUN dnf --releasever=$(rpm -q system-release --qf '%{VERSION}') \
  --installroot /sysroot \
  -y \
  --setopt=install_weak_deps=False \
```

```
install bash

FROM scratch
COPY --from=build /sysroot /
WORKDIR /
ENTRYPOINT ["/bin/bash"]
```

- Questo Dockerfile funziona tramite:

1. Avvio di un contenitore denominato `AL2023 . build`. Questo container viene utilizzato per il bootstrap dei container essenziali; non viene implementato, ma genera il container da implementare.
2. Creazione della directory `/sysroot`. In questa directory il container `build` installa le dipendenze necessarie per il container essenziale. In un passaggio successivo, il percorso `/sysroot` viene impacchettato per diventare la directory principale dell'immagine essenziale.

Usare l'opzione `--installroot` di `dnf` in questo modo è il modo in cui creiamo le altre `AL2023` immagini. Si tratta di una funzionalità di `dnf` che consente il funzionamento dei programmi di installazione e degli strumenti per la creazione di immagini.

3. Richiamo di `dnf` per installare pacchetti in `/sysroot`.

Il comando `rpm -q system-release --qf '%{VERSION}'` interroga (`-q`) il pacchetto `system-release`, impostando il formato della query (`--qf`) per stampare la versione del pacchetto sottoposta a query (la variabile `%{VERSION}` è la variabile `rpm` per la versione di RPM).

Impostando l'argomento `--releasever` di `dnf` sulla versione di `system-release` nel container `build`, il Dockerfile può essere utilizzato per creare nuovamente il container essenziali ogni volta che viene rilasciata un'immagine di container di base aggiornata di Amazon Linux.

È possibile impostarlo su qualsiasi versione di Amazon Linux 2023, `--releasever` ad esempio `2023.11.20260427`. Ciò significherebbe che il `build` contenitore funzionerebbe come `AL2023` versione più recente, ma creerebbe il contenitore `barebones` a partire dalla versione `2023.11.20260427` indipendentemente dalla versione corrente. `AL2023`

L'opzione di configurazione `--setopt=install_weak_deps=False` istruisce `dnf` affinché installi solo le dipendenze che sono richieste, anziché quelle consigliate o suggerite.

4. Copia del sistema installato nella directory principale di un container vuoto (FROM scratch).
 5. Impostazione di `ENTRYPOINT` come binario desiderato, in questo caso `/bin/bash`.
3. Crea una directory vuota e aggiungi il contenuto dell'esempio riportato nel passaggio 2 a un file denominato `Dockerfile`.

```
$ mkdir al2023-barebones-bash-example
$ cd al2023-barebones-bash-example
$ cat > Dockerfile <<EOF
FROM public.ecr.aws/amazonlinux/amazonlinux:2023 as build
RUN mkdir /sysroot
RUN dnf --releasever=$(rpm -q system-release --qf '%{VERSION}') \
  --installroot /sysroot \
  -y \
  --setopt=install_weak_deps=False \
  install bash && dnf --installroot /sysroot clean all

FROM scratch
COPY --from=build /sysroot /
WORKDIR /
ENTRYPOINT ["/bin/bash"]
EOF
```

4. Crea il container eseguendo il comando seguente.

```
$ docker build -t al2023-barebones-bash-example
```

5. Esegui il container usando il comando seguente per capire quanto è minimo un container solo `bash`.

```
$ docker run -it --rm al2023-barebones-bash-example
bash-5.2# rpm
bash: rpm: command not found
bash-5.2# du -sh /usr/
bash: du: command not found
bash-5.2# ls
```

```
bash: ls: command not found
bash-5.2# echo /bin/*
/bin/alias /bin/bash /bin/bashbug /bin/bashbug-64 /bin/bg /bin/catchsegv /bin/cd /
bin/command /bin/fc /bin/fg /bin/gencat /bin/getconf /bin/getent /bin/getopts /
bin/hash /bin/iconv /bin/jobs /bin/ld.so /bin/ldd /bin/locale /bin/localedef /
bin/pldd /bin/read /bin/sh /bin/sotruss /bin/sprof /bin/type /bin/tzselect /bin/
ulimit /bin/umask /bin/unalias /bin/wait /bin/zdump
```

Per un esempio più pratico, la procedura seguente consente di creare un container per un'applicazione C che visualizza Hello World!.

1. Crea una directory vuota, quindi aggiungi il codice sorgente C e Dockerfile.

```
$ mkdir al2023-barebones-c-hello-world-example
$ cd al2023-barebones-c-hello-world-example
$ cat > hello-world.c <<EOF
#include <stdio.h>
int main(void)
{
    printf("Hello World!\n");
    return 0;
}
EOF

$ cat > Dockerfile <<EOF
FROM public.ecr.aws/amazonlinux/amazonlinux:2023 as build
COPY hello-world.c /
RUN dnf -y install gcc
RUN gcc -o hello-world hello-world.c
RUN mkdir /sysroot
RUN mv hello-world /sysroot/
RUN dnf --releasever=$(rpm -q system-release --qf '%{VERSION}') \
    --installroot /sysroot \
    -y \
    --setopt=install_weak_deps=False \
    install glibc && dnf --installroot /sysroot clean all

FROM scratch
COPY --from=build /sysroot /
WORKDIR /
ENTRYPOINT ["/hello-world"]
EOF
```

2. Avvia il container utilizzando il comando seguente.

```
$ docker build -t al2023-barebones-c-hello-world-example .
```

3. Esegui il container utilizzando il comando seguente.

```
$ docker run -it --rm al2023-barebones-c-hello-world-example  
Hello World!
```

Confronto dei pacchetti installati sulle immagini di container Amazon Linux 2023

Un confronto tra il RPMs presente sull'immagine del contenitore di AL2023 base e il RPMs presente sull'immagine AL2023 minima del contenitore.

Pacchetto	Contenitore	Contenitore minimo
alternatives	1.15	1.15
amazon-linux-repo-cdn	20236.20241031	2023,620241031
audit-libs	3,0,6	3.0.6
basesystem	11	11
bash	5,2,15	5,2,15
bzip2-libs	1.0.8	1.0.8
ca-certificates	2023,2,68	2023,2,68
coreutils-single	8,32	8,32
crypto-policies	20220428	20220428

Pacchetto	Contenitore	Contenitore minimo
curl-minimal	8,5,0	8,5,0
dnf	4.14.0	
dnf-data	4.14.0	4.14.0
elfutils-default-yama-scope	0.188	
elfutils-libelf	0.188	
elfutils-libs	0.188	
expat	2.5.0	
file-libs	5,39	5,39
filesystem	3,14	3,14
gawk	5.1.0	5.1.0
gdbm-libs	1,19	
glib2	2,74,7	2,74,7
glibc	2,34	2,34
glibc-common	2,34	2,34
glibc-minimal-langpack	2,34	2,34
gmp	62,1	6.2.1
gnupg2-minimal	2.3.7	2.3.7
gobject-introspection		1,73,0

Pacchetto	Contenitore	Contenitore minimo
gpgme	1.15.1	1.15.1
grep	3.8	3.8
json-c	0,14	0,14
keyutils-libs	1.6.3	1.6.3
krb5-libs	1,21,3	1,21,3
libacl	2.3.1	2.3.1
libarchive	3,7,4	3,7,4
libassuan	2,5,5	2,5,5
libattr	2.5.1	2.5.1
libblkid	2,37,4	2,37,4
libcap	2,48	2,48
libcap-ng	0.8.2	0.8.2
libcom_err	1,446,5	1,46,5
libcomps	01,20	
libcurl-minimal	8,5,0	8,5,0
libdnf	0,69,0	0,69,0
libffi	34.4	34.4
libgcc	114,1	11.4.1
libgcrypt	1.10.2	1.10.2
libgomp	11.4.1	

Pacchetto	Contenitore	Contenitore minimo
libgpg-error	1,42	1,42
libidn2	2.3.2	2.3.2
libmodulemd	2.13.0	2.13.0
libmount	2,37,4	2,37,4
libnghttp2	1,59,0	1,59,0
libpeas		1.32.0
libpsl	0,21,1	0,21,1
librepo	1,14,5	1,14,5
libreport-filessystem	2,15,2	2,15,2
libselinux	3.4	3.4
libsepol	3.4	3.4
libsigsegv	2,13	2,13
libsmartcols	2,37,4	2,37,4
libsolv	0,7,22	0,7,22
libstdc++	11,4,1	11.4.1
libtasn1	4,19,0	4,19,0
libunistring	0,9,10	0,9,10
libuuid	2,37,4	2,37,4
libverto	0,32	0,32
libxcrypt	4,4,33	

Pacchetto	Contenitore	Contenitore minimo
libxml2	2,10,4	210.4
libyaml	02,5	02,5
libzstd	1,5,5	1,5,5
lua-libs	5.4.4	5.4.4
lz4-libs	1.9.4	1.9.4
microdnf		3.10.0
microdnf-dnf		3.10.0
mpfr	4.1.0	4.1.0
ncurses-base	6.2	6.2
ncurses-libs	6.2	6.2
npth	1.6	1.6
openssl-libs	30,8	30,8
p11-kit	0,24,1	0,24,1
p11-kit-trust	0,24,1	0,24,1
pcre2	10,40	10,40
pcre2-syntax	10,40	10,40
popt	1,18	1,18
publicsuffix-list-dafsa	20240212	20240212
python3	3,9,16	
python3-dnf	4.14.0	

Pacchetto	Contenitore	Contenitore minimo
python3-gpg	1.15.1	
python3-hawkey	0,69,0	
python3-libcomps	01,20	
python3-libdnf	0,69,0	
python3-libs	3,9,16	
python3-pip-wheel	21,31	
python3-rpm	4,161,3	
python3-setuptools-wheel	59,60	
readline	8.1	8.1
rpm	4,161,3	4,161,3
rpm-build-libs	4,161,3	
rpm-libs	4,161,3	4,161,3
rpm-sign-libs	4,161,3	
sed	4.8	4.8
setup	2,13,7	2,13,7
sqlite-libs	3,4,0	3,4,0
system-release	2023,620241031	2023,620241031
tzdata	2024a	
xz-libs	5.2.5	5.2.5
yum	4.14.0	

Pacchetto	Contenitore	Contenitore minimo
zlib	1,2,11	1,2,11

Confronto dei pacchetti installati sulle immagini di container e AMI minime Amazon Linux 2023

Un confronto tra il RPMs presente sull'AMI AL2023 minimale e il RPMs presente sulle immagini di AL2023 base e minimali del contenitore.

Pacchetto	AMI minima	Contenitore	Contenitore minimo
alternatives	1.15	1.15	1.15
amazon-chrony-config	4.3		
amazon-ec2-net-utils	2.5.1		
amazon-linux-repo-cdn		20236.20241031	2023,620241031
amazon-linux-repo-s3	2023,620241031		
amazon-linux-sb-keys	2023,1		
amd-ucode-firmware	20210208 (marzo)		
audit	3.0.6		
audit-libs	30.6	30.6	30.6
awscli-2	2,15,30		

Pacchetto	AMI minima	Contenitore	Contenitore minimo
basesystem	11	11	11
bash	5,2,15	5,2,15	5,2,15
bzip2-libs	1.0.8	1.0.8	1.0.8
ca-certificates	2023,2,68	2023,2,68	2023,2,68
checkpolicy	3.4		
chrony	4.3		
cloud-init	222,2		
cloud-init-cfg-ec2	222,2		
cloud-utils-growpart	0,31		
coreutils	8,32		
coreutils-common	8,32		
coreutils-single		8,32	8,32
cpio	2,13		
cracklib	2,9,6		
cracklib-dicts	29.6		
crypto-policies	20220428	20220428	20220428
cryptsetup-libs	2.6.1		
curl-minimal	8,5,0	8,5,0	8,5,0

Pacchetto	AMI minima	Contenitore	Contenitore minimo
cyrus-sasl-lib	2,1,27		
dbus	1,12,28		
dbus-broker	32		
dbus-common	1,12,28		
dbus-libs	1,12,28		
device-mapper	1,02,185		
device-mapper-libs	1,02,185		
diffutils	3.8		
dnf	4.14.0	4.14.0	
dnf-data	4.14.0	4.14.0	4.14.0
dnf-plugin-release-notification	1.2		
dnf-plugins-core	4.3.0		
dnf-plugin-support-info	1.2		
dracut	055		
dracut-config-ec2	3.0		
dracut-config-generic	055		

Pacchetto	AMI minima	Contenitore	Contenitore minimo
e2fsprogs	1,446,5		
e2fsprogs-libs	1,46,5		
ec2-utils	2.2.0		
efi-filesystem	5		
efivar	38		
efivar-libs	38		
elfutils- default-yama- scope	0.188	0.188	
elfutils-libelf	0.188	0.188	
elfutils-libs	0.188	0.188	
expat	2.5.0	2.5.0	
file	5,39		
file-libs	5,39	5,39	5,39
filesystem	3,14	3,14	3,14
findutils	4.8.0		
fuse-libs	2,9,9		
gawk	5.1.0	5.1.0	5.1.0
gdbm-libs	1,19	1,19	
gdisk	1.0.8		
gettext	0,21		

Pacchetto	AMI minima	Contenitore	Contenitore minimo
gettext-libs	0,21		
glib2	2,74,7	2,74,7	2,74,7
glibc	2,34	2,34	2,34
glibc-all-langpacks	2,34		
glibc-common	2,34	2,34	2,34
glibc-locale-source	2,34		
glibc-minimal-langpack		2,34	2,34
gmp	62,1	6.2.1	6.2.1
gnupg2-minimal	2.3.7	2.3.7	2.3.7
gnutls	3.8.0		
gobject-introspection			1,73,0
gpgme	1.15.1	1.15.1	1.15.1
grep	3.8	3.8	3.8
groff-base	1,22,4		
grub2-common	2,06		
grub2-efi-aa64-ec2	2,06 (aarch64)		
grub2-efi-x64-ec2	2,06 (x86_64)		

Pacchetto	AMI minima	Contenitore	Contenitore minimo
grub2-pc-modules	2,06		
grub2-tools	2,06		
grub2-tools-minimal	2,06		
grubby	8,40		
gzip	1.12		
hostname	3,23		
hwdata	0,384		
inih	49		
initscripts	10,09		
iproute	6.10.0		
iputils	20210202		
irqbalance	1.9.0		
jansson	2.14		
jitterentropy	3.4.1		
jq	1.7.1		
json-c	0,14	0,14	0,14
kbd	2.4.0		
kbd-misc	2.4.0		
kernel	6,1112		

Pacchetto	AMI minima	Contenitore	Contenitore minimo
kernel-libbpf	6,1112		
kernel- livepatch-repo- s3	2023,620241031		
keyutils-libs	1.6.3	1.6.3	1.6.3
kmod	29		
kmod-libs	29		
krb5-libs	1,21,3	1,21,3	1,21,3
less	608		
libacl	2.3.1	2.3.1	2.3.1
libarchive	3,7,4	3,7,4	3,7,4
libargon2	20171227		
libassuan	2,5,5	2,5,5	2,5,5
libattr	2.5.1	2.5.1	2.5.1
libblkid	2,37,4	2,37,4	2,37,4
libcap	2,48	2,48	2,48
libcap-ng	0.8.2	0.8.2	0.8.2
libcbor	0.7.0		
libcom_err	1,446,5	1,46,5	1,46,5
libcomps	01,20	01,20	
libcurl-minimal	8,5,0	8,5,0	8,5,0

Pacchetto	AMI minima	Contenitore	Contenitore minimo
libdb	5,3,28		
libdnf	0,69,0	0,69,0	0,69,0
libeconf	0,40		
libedit	3.1		
libfdisk	2,37,4		
libffi	34.4	34.4	34.4
libfido2	1.10.0		
libgcc	114,1	11.4.1	11.4.1
libgcrypt	1.10.2	1.10.2	1.10.2
libgomp	11.4.1	11.4.1	
libgpg-error	1,42	1,42	1,42
libidn2	2.3.2	2.3.2	2.3.2
libkcapi	1.4.0		
libkcapi-hmacalc	1.4.0		
libmnl	1.0.4		
libmodulemd	2.13.0	2.13.0	2.13.0
libmount	2,37,4	2,37,4	2,37,4
libnghttp2	1,59,0	1,59,0	1,59,0
libpeas			1.32.0
libpipeline	1.5.3		

Pacchetto	AMI minima	Contenitore	Contenitore minimo
libpsl	0,21,1	0,21,1	0,21,1
libpwquality	1.4.4		
librepo	1,14,5	1,14,5	1,14,5
libreport-filesystem	2,15,2	2,15,2	2,15,2
libseccomp	2.5.3		
libselinux	3.4	3.4	3.4
libselinux-utils	3.4		
libsemanage	3.4		
libsepol	3.4	3.4	3.4
libsigsegv	2,13	2,13	2,13
libsmartcols	2,37,4	2,37,4	2,37,4
libsolv	0,7,22	0,7,22	0,7,22
libss	1,446,5		
libstdc++	11.4.1	11.4.1	11.4.1
libtasn1	4,19,0	4,19,0	4,19,0
libtextstyle	0,21		
libunistring	0,9,10	0,9,10	0,9,10
libuser	0,63		
libutempter	1.2.1		

Pacchetto	AMI minima	Contenitore	Contenitore minimo
libuuid	2,37,4	2,37,4	2,37,4
libverto	0,32	0,32	0,32
libxcrypt	4,4,33	4,4,33	
libxml2	2,10,4	210.4	210.4
libyaml	02,5	02,5	02,5
libzstd	1,5,5	1,5,5	1,5,5
linux-firmware-whence	20210208 (novembre)		
logrotate	3.20.1		
lua-libs	5.4.4	5.4.4	5.4.4
lz4-libs	1.9.4	1.9.4	1.9.4
man-db	2,9,3		
microcode_ctl	2.1 (x86_64)		
microdnf			3.10.0
microdnf-dnf			3.10.0
mpfr	4.1.0	4.1.0	4.1.0
ncurses	6.2		
ncurses-base	6.2	6.2	6.2
ncurses-libs	6.2	6.2	6.2
nettle	3.8		
net-tools	2.0		

Pacchetto	AMI minima	Contenitore	Contenitore minimo
npth	1.6	1.6	1.6
numactl-libs	2,0,14		
oniguruma	6,9,7,1		
openldap	2,4,57		
openssh	8.7p1		
openssh-clients	8,7p1		
openssh-server	8,7p1		
openssl	3,0,8		
openssl-libs	30,8	30,8	30,8
openssl-pkcs11	0,4,12		
os-prober	1,77		
p11-kit	0,24,1	0,24,1	0,24,1
p11-kit-trust	0,24,1	0,24,1	0,24,1
pam	1.5.1		
passwd	0,80		
pciutils	3.7.0		
pciutils-libs	3.7.0		
pcre2	10,40	10,40	10,40
pcre2-syntax	10,40	10,40	10,40
policycoreutils	3.4		

Pacchetto	AMI minima	Contenitore	Contenitore minimo
popt	1,18	1,18	1,18
procps-ng	3,3,17		
psmisc	23,4		
publicsuffix- list-dafsa	20240212	20240212	20240212
python3	3,9,16	3,9,16	
python3-attrs	20,3,0		
python3-audit	30.6		
python3-awscrt	0,19,19		
python3-babel	2,9,1		
python3-cffi	1,14,5		
python3-chardet	4.0.0		
python3-c olorama	04.4		
python3-c onfigobj	5.0.6		
python3-c ryptography	36,0		
python3-d ateutil	28.1		
python3-dbus	1,2,18		
python3-distro	1.5.0		

Pacchetto	AMI minima	Contenitore	Contenitore minimo
python3-dnf	4.14.0	4.14.0	
python3-dnf- plugins-core	4.3.0		
python3-d ocutils	0,16		
python3-gpg	1.15.1	1.15.1	
python3-hawkey	0,69,0	0,69,0	
python3-idna	(2.10)		
python3-jinja2	211,3		
python3-j mespath	0.10.0		
python3-j sonpatch	1,21		
python3-j sonpointer	2.0		
python3-j sonschema	3.2.0		
python3-l ibcomps	01,20	01,20	
python3-libdnf	0,69,0	0,69,0	
python3-libs	3,9,16	3,9,16	
python3-l ibselinux	3.4		

Pacchetto	AMI minima	Contenitore	Contenitore minimo
python3-l ibsemanage	3.4		
python3-m arkupsafe	1.1.1		
python3-n etifaces	0,10,6		
python3-o authlib	3.0.2		
python3-pip- wheel	21,31	21,31	
python3-ply	3,11		
python3-p olicycoreutils	3.4		
python3-p rettytable	0.7.2		
python3-prompt- toolkit	3,0,24		
python3-p ycparser	2,20		
python3-p yrsistent	0,17,3		
python3-p yserial	3.4		
python3-pysocks	1.7.1		
python3-pytz	2022,7,1		

Pacchetto	AMI minima	Contenitore	Contenitore minimo
python3-pyyaml	5.4.1		
python3-requests	2,25,1		
python3-rpm	4,161,3	4,161,3	
python3-ruamel-yaml	0,16,6		
python3-ruamel-yaml-clib	0,12		
python3-setuptools	4.4.1		
python3-setuptools	59,60		
python3-setuptools-wheel	59,6,0	59,6,0	
python3-six	1.15.0		
python3-systemd	235		
python3-urllib3	1,25,10		
python3-wcwidth	0,2,5		
readline	8.1	8.1	8.1
rng-tools	6,14		
rootfiles	8.1		
rpm	4,161,3	4,161,3	4,161,3
rpm-build-libs	4,161,3	4,161,3	

Pacchetto	AMI minima	Contenitore	Contenitore minimo
rpm-libs	4,161,3	4,161,3	4,161,3
rpm-plugin-selinux	4,161,3		
rpm-plugin-systemd-inhibit	4,161,3		
rpm-sign-libs	4,161,3	4,161,3	
sbsigntools	0.9.4		
sed	4.8	4.8	4.8
selinux-policy	381,45		
selinux-policy-targeted	381,45		
setup	2,13,7	2,13,7	2,13,7
shadow-utils	4.9		
sqlite-libs	3,4,0	3,4,0	3,4,0
sudo	1,9,15		
sysctl-defaults	1.0		
systemd	252,23		
systemd-libs	252,23		
systemd-networkd	252,23		
systemd-pam	252,23		

Pacchetto	AMI minima	Contenitore	Contenitore minimo
systemd-resolved	252,23		
systemd-udev	252,23		
system-release	2023,620241031	2023,620241031	2023,620241031
tar	1,34		
tzdata	2024a	2024a	
update-motd	2.2		
userspace-rcu	0.12.1		
util-linux	2,37,4		
util-linux-core	2,37,4		
vim-data	9,0,2153		
vim-minimal	9,0,2153		
which	2,21		
xfspgrog	5,18,0		
xz	5.2.5		
xz-libs	5.2.5	5.2.5	5.2.5
yum	4.14.0	4.14.0	
zlib	1,2,11	1,2,11	1,2,11
zram-generator	1.1.2		
zram-generator-defaults	1.1.2		

Pacchetto	AMI minima	Contenitore	Contenitore minimo
zstd	1,5,5		

AL2023 su AWS Elastic Beanstalk

AWS Elastic Beanstalk è un servizio per la distribuzione e la scalabilità di applicazioni e servizi Web. Carica il codice ed Elastic Beanstalk gestirà automaticamente l'implementazione, dal provisioning della capacità, il sistema di bilanciamento del carico e il dimensionamento automatico, al monitoraggio dello stato delle applicazioni. Per ulteriori informazioni, consulta [AWS Elastic Beanstalk](#).

Per usare Elastic Beanstalk, devi creare un'applicazione, caricare una versione dell'applicazione sotto forma di un bundle di origine dell'applicazione (ad esempio un file .war Java) in Elastic Beanstalk e fornire alcune informazioni sull'applicazione. Elastic Beanstalk avvia automaticamente un ambiente e crea e AWS configura le risorse necessarie per eseguire il codice. Per ulteriori informazioni, consulta la [Guida per gli sviluppatori di AWS Elastic Beanstalk](#).

Le piattaforme Linux Elastic Beanstalk EC2 utilizzano istanze Amazon e queste istanze eseguono Amazon Linux. A partire dal 4 agosto 2023, Elastic Beanstalk offre le seguenti ramificazioni della piattaforma basati su Amazon Linux 2023: Docker, Tomcat, Java SE, Node.js, PHP e Python. Elastic Beanstalk sta lavorando per rilasciare il supporto per AL2 023 su altre piattaforme Elastic Beanstalk.

<https://docs.aws.amazon.com/elasticbeanstalk/latest/dg/Welcome> L'elenco completo delle piattaforme Elastic Beanstalk supportate e delle piattaforme attuali basate su 023 è disponibile nella sezione Piattaforme AL2 Elastic Beanstalk Linux della [Elastic Beanstalk Developer Guide](#).

Puoi trovare le note di rilascio per le nuove piattaforme Elastic Beanstalk e le versioni delle piattaforme esistenti nelle [note di rilascio di Elastic Beanstalk](#).

Usando AL2 0,23 pollici AWS CloudShell

AWS CloudShell è una shell preautenticata basata su browser che puoi avviare direttamente da Console di gestione AWS. È possibile accedere CloudShell da diversi modi Console di gestione AWS. Per ulteriori informazioni, vedi [Come iniziare AWS CloudShell?](#)

AWS CloudShell, che attualmente è basato su Amazon Linux 2, migrerà alla versione AL2 023. La migrazione a AL2 023 inizierà a essere implementata in tutti i casi a Regioni AWS partire dal 4

dicembre 2023. Per ulteriori informazioni sulla CloudShell migrazione a AL2 023, consulta [AWS CloudShell Migrazione da Amazon Linux 2 ad Amazon Linux 2023](#).

Utilizzo di Amazon ECS basato su AL2 023 AMIs per ospitare carichi di lavoro containerizzati

Note

Per ulteriori informazioni su come utilizzare AL2 023 all'interno di un contenitore, consulta [AL2023 in contenitori](#).

Amazon Elastic Container Service (Amazon ECS) è un servizio di orchestrazione di container completamente gestito che facilita l'implementazione, la gestione e il dimensionamento delle applicazioni containerizzate. Essendo un servizio completamente gestito, Amazon ECS include AWS configurazioni e best practice operative integrate. È integrato con strumenti AWS sia di terze parti, come Amazon Elastic Container Registry (Amazon ECR) e Docker. Questa integrazione consente ai team di concentrarsi più facilmente sulla creazione delle applicazioni piuttosto che sull'ambiente. Puoi eseguire e dimensionare i carichi di lavoro dei container nelle regioni AWS nel cloud, senza la complessità legata alla gestione di un piano di controllo (control-plane).

Puoi ospitare carichi di lavoro containerizzati su AL2 023 utilizzando l'AMI ottimizzata per AL2 Amazon ECS basata su 023. Per ulteriori informazioni, consulta l'AMI ottimizzata per [Amazon ECS](#)

Modifiche nello AL2 023 per Amazon ECS rispetto a AL2

Analogamente AL2, AL2 023 fornisce i pacchetti di base necessari per l'esecuzione come istanza Amazon ECS Linux. Incontainerd,docker, AL2 i ecs-init pacchetti erano disponibili tramiteamazon-linux-extras, mentre AL2 023 include questi pacchetti nei repository principali.

Con gli aggiornamenti deterministici tramite la funzionalità di repository con versioni, ogni AL2 AMI 023 per impostazione predefinita è bloccata su una versione del repository specifica. Questo vale anche per l'AMI ottimizzata Amazon ECS AL2 023. Tutti gli aggiornamenti dell'ambiente possono essere gestiti e testati con attenzione prima della distribuzione, oltre a fornire un modo semplice per ripristinare il contenuto di un'AMI precedente in caso di problemi. Per ulteriori informazioni su questa funzionalità AL2 023, vedere. [Aggiornamenti deterministici tramite repository con versioni su AL2023](#)

AL2023 passa a cgroup v2 tramite l'interfaccia cgroup v1 supportata in. AL2 Per ulteriori informazioni, consulta [Gerarchia dei gruppi di controllo unificati \(cgroup v2\)](#).

Note

AL2Le versioni 023 precedenti alla [2023.2.20230920 \(la prima versione AL2 023.2\)](#) contenevano un bug nella gestione di (OOM) all'interno di un cgroup. systemd Out-of-Memory Tutti i processi in cgroup venivano sempre interrotti invece che OOM-Killer scegliesse un processo alla volta, che è il comportamento previsto.

Si trattava di una regressione rispetto al AL2 comportamento ed è stata risolta a partire dalla versione 2023.2.20230920 di 023. AL2

[Il codice per creare l'AMI ottimizzata per Amazon ECS è disponibile nel amazon-ecs-ami GitHub progetto](#). Le [note di rilascio](#) descrivono quale versione AL2 023 è mappata a quale versione AMI Amazon ECS.

Personalizzazione dell'AMI ottimizzata per Amazon AL2 ECS basata su 023

Important

Ti consigliamo di utilizzare l' AL2AMI 023 ottimizzata per Amazon ECS. Per ulteriori informazioni, consulta l'[AMI ottimizzata per Amazon ECS](#) nella Amazon Elastic Container Service Developer Guide.

Puoi utilizzare gli stessi script di compilazione utilizzati da Amazon ECS per creazioni personalizzate. AMIs Per ulteriori informazioni, consulta lo script di [compilazione dell'AMI Linux ottimizzato per Amazon ECS](#).

Utilizzo di Amazon Elastic File System su AL2 023

Amazon Elastic File System (Amazon EFS) fornisce un'archiviazione di file serverless e completamente elastica in modo da poter condividere i dati dei file senza dover fornire o gestire la capacità e le prestazioni di archiviazione. Amazon EFS è progettato per eseguire il dimensionamento on-demand fino a svariati petabyte senza interrompere le applicazioni, aumentando e riducendo automaticamente le dimensioni man mano che aggiungi e rimuovi i file. Poiché Amazon EFS dispone di una semplice interfaccia di servizi Web, è possibile creare e configurare i file system in modo

rapido e semplice. Il servizio gestisce tutta l'infrastruttura di storage dei file per conto dell'utente, il che significa che è possibile evitare attività complesse come la distribuzione, l'applicazione di patch e la manutenzione di complesse configurazioni di file system.

Amazon EFS supporta il protocollo Network File System versione 4 (NFSv4.1 e NFSv4 .0), quindi le applicazioni e gli strumenti che usi oggi funzionano perfettamente con Amazon EFS. Più istanze di elaborazione, tra cui EC2 Amazon, Amazon ECS e AWS Lambda, possono accedere contemporaneamente a un file system Amazon EFS. Di conseguenza, un file system EFS può fornire un'origine dati comune per carichi di lavoro e applicazioni in esecuzione su più server o istanze di calcolo.

Installazione su 023 **amazon-efs-utils** AL2

Il `amazon-efs-utils` pacchetto è disponibile nei repository AL2 023 per essere installato e utilizzato per accedere ai file system Amazon EFS.

Installa il **amazon-efs-utils** pacchetto su 023 AL2

- Installa `amazon-efs-utils` usando il seguente comando.

```
$ dnf -y install amazon-efs-utils
```

Montaggio di un file system Amazon EFS su AL2 023

Dopo `amazon-efs-utils` l'installazione, puoi montare un file system Amazon EFS sulla tua istanza AL2 023.

Monta un file system Amazon EFS su AL2 023

- Per eseguire il montaggio utilizzando l'id del file system, usa il seguente comando.

```
sudo mount -t efs file-system-id efs-mount-point/
```

Puoi eseguire anche il montaggio del file system in modo che i dati in transito siano crittografati utilizzando TLS oppure il nome DNS o l'IP di destinazione di montaggio anziché l'id del file system. Per ulteriori informazioni, consulta la pagina dedicata al [montaggio su istanze Amazon Linux utilizzando l'assistente per il montaggio EFS](#).

Utilizzo di Amazon EMR integrato AL2023

Amazon EMR è un servizio Web che rende più semplice ed efficiente l'elaborazione di grandi quantità di dati utilizzando Apache Hadoop e i servizi offerti da AWS.

AL2023 versioni basate su Amazon EMR

La release 7.0.0 di Amazon EMR è stata la prima versione basata su AL2023. Con questa versione, AL2023 è il sistema operativo di base per Amazon EMR, che offre tutti i vantaggi di AL2023 Amazon EMR. Per ulteriori informazioni, consulta le note di rilascio di [Amazon EMR 7.0.0](#).

AL2023 basato su Amazon EMR su EKS

Amazon EMR su EKS 6.13 è stata la prima release introdotta AL2023 come opzione. Con questa versione, puoi avviare Spark AL2023 come sistema operativo, insieme al runtime Java 17. Per ulteriori informazioni, consulta le note di rilascio di [Amazon EMR su EKS 6.13 e tutte le note di rilascio](#) di Amazon [EMR](#) su EKS.

Utilizzo AL2023 in AWS Lambda

Con AWS Lambda, puoi eseguire codice senza effettuare il provisioning o gestire server. Verrà addebitato soltanto il tempo di calcolo utilizzato e non verrà addebitato alcun costo quando il codice non è in esecuzione. Puoi eseguire codice per qualsiasi tipo di applicazione o servizio di back-end, senza alcuna amministrazione. Basta caricare il codice e Lambda penserà a tutto ciò che serve per eseguirlo e dimensionarlo con alta disponibilità.

AL2023 **provided.al2023** runtime gestito e immagine del contenitore

Il runtime di `provided.al2023` base si basa sull'[immagine AL2023 minima del contenitore](#) e fornisce un runtime gestito da Lambda e un'immagine di [base del contenitore AL2023](#) basati su Lambda. Poiché il `provided.al2023` runtime si basa sull'immagine AL2023 minima del contenitore, è sostanzialmente più piccolo (meno di 40 MB) rispetto al `provided.al2` runtime a circa 109 MB.

Per ulteriori informazioni, consulta [Lambda runtimes](#) e Working [with Lambda container images](#).

AL2023 runtime Lambda basati

Le versioni future dei runtime linguistici gestiti, come Node.js 20, Python 3.12, Java 21 e .NET 8, si basano AL2023 e verranno utilizzate `provided.al2023` come immagine di base come descritto nell'[annuncio dei AL2023](#) runtime basati.

AL2023 funzioni Lambda basate su

- [AL2023 Funzioni Lambda scritte in Go](#)
- [AL2023 Funzioni Lambda scritte in Rust](#)

Per ulteriori informazioni, consulta [Lambda runtimes](#) nella Developer Guide.AWS Lambda

Esercitazioni

I seguenti tutorial mostrano come eseguire attività comuni utilizzando istanze Amazon EC2 che eseguono Amazon Linux 2023 (). AL2023 [Per i tutorial video, consulta Video didattici e laboratori.AWS](#)

Per AL2 istruzioni, consulta [i tutorial per le istanze Amazon EC2 che eseguono Linux nella Amazon EC2 User Guide](#).

Esercitazioni

- [Tutorial: Installare un server LAMP su AL2023](#)
- [Tutorial: Configura SSL/TLS su AL2023](#)
- [Tutorial: Ospita un WordPress blog su AL2023](#)
- [Tutorial: transizione da Redis 6 a Valkey su 023 AL2](#)
- [Tutorial: Installare l'ambiente desktop GNOME su AL2023](#)
- [Tutorial: configurare il server TigerVNC su AL2023](#)
- [Utilizzo di Multi-Gen LRU \(MGLRU\) sui kernel AL2023](#)
- [Tutorial: Installazione LibreOffice su AL2023](#)

Tutorial: Installare un server LAMP su AL2023

[Le seguenti procedure consentono di installare un server Web Apache con supporto PHP e MariaDB \(un fork di MySQL sviluppato dalla comunità\) sull'istanza \(a volte chiamata server web LAMP o stack LAMP\).](#) AL2023 Puoi usare questo server per ospitare un sito Web statico o distribuire un'applicazione PHP dinamica che legge e scrive informazioni in un database.

Important

Queste procedure sono destinate all'uso con AL2023. Se si sta tentando di configurare un server web LAMP su una distribuzione diversa, come Ubuntu o Red Hat Enterprise Linux, questo tutorial non funzionerà. Per Ubuntu, consulta la seguente documentazione della comunità Ubuntu: [ApacheMySQLPHP](#). Per altre distribuzioni, consulta la relativa documentazione specifica.

Processi

- [Fase 1: preparare il server LAMP](#)
- [Fase 2: verificare il server LAMP](#)
- [Fase 3: proteggere il server di database](#)
- [Fase 4: Installazione \(facoltativa\) phpMyAdmin](#)
- [Risoluzione dei problemi](#)
- [Argomenti correlati](#)

Fase 1: preparare il server LAMP

Prerequisiti

- Questo tutorial presuppone che tu abbia già lanciato una nuova istanza utilizzando AL2023, con un nome DNS pubblico raggiungibile da Internet. Per ulteriori informazioni, consulta [AL2023 su Amazon EC2](#). È inoltre necessario aver configurato il gruppo di sicurezza per consentire le connessioni SSH (porta 22), HTTP (porta 80) e HTTPS (porta 443). Per ulteriori informazioni su questi prerequisiti, consulta [Autorizza il traffico in entrata per le tue istanze Linux nella Amazon EC2 User Guide](#).
- La procedura seguente installa l'ultima versione di PHP disponibile su, attualmente 8.1. AL2023 Se hai in programma di utilizzare applicazioni PHP diverse da quelle descritte in questo tutorial, devi verificare che siano compatibili con la versione 8.1.

Per preparare il server LAMP

1. Connettiti alla tua istanza. Per ulteriori informazioni, consulta [AL2023 Connessione alle istanze](#).
2. Per verificare che tutti i pacchetti software siano aggiornati, eseguire un aggiornamento rapido del software sull'istanza. Questo processo può richiedere alcuni minuti, ma è importante accertarsi che siano disponibili gli aggiornamenti della sicurezza e le correzioni dei bug più recenti.

L'opzione `-y` installa gli aggiornamenti senza chiedere conferma. Se desideri esaminare gli aggiornamenti prima di installarli, puoi omettere questa opzione.

```
[ec2-user ~]$ sudo dnf upgrade -y
```

3. Installa le versioni più recenti del server web Apache e dei pacchetti PHP per. AL2023

```
[ec2-user ~]$ sudo dnf install -y httpd wget php-fpm php-mysql php-json php php-devel
```

4. Installa i pacchetti software MariaDB. Utilizzare il comando `dnf install` per installare contemporaneamente più pacchetti software e tutte le dipendenze correlate.

```
[ec2-user ~]$ sudo dnf install mariadb105-server
```

È possibile visualizzare le versioni correnti di tali pacchetti utilizzando il comando seguente:

```
[ec2-user ~]$ sudo dnf info package_name
```

Esempio:

```
[root@ip-172-31-25-170 ec2-user]# dnf info mariadb105
Last metadata expiration check: 0:00:16 ago on Tue Feb 14 21:35:13 2023.
Installed Packages
Name           : mariadb105
Epoch         : 3
Version        : 10.5.16
Release        : 1.amzn2023.0.6
Architecture   : x86_64
Size           : 18 M
Source         : mariadb105-10.5.16-1.amzn2023.0.6.src.rpm
Repository     : @System
From repo      : amazonlinux
Summary        : A very fast and robust SQL database server
URL            : http://mariadb.org
License        : GPLv2 and LGPLv2
Description    : MariaDB is a community developed fork from MySQL - a multi-user,
multi-threaded
                : SQL database server. It is a client/server implementation consisting
of
                : a server daemon (mariadb) and many different client programs and
libraries.
                : The base package contains the standard MariaDB/MySQL client programs
and
                : utilities.
```

5. Avviare il server Web Apache.

```
[ec2-user ~]$ sudo systemctl start httpd
```

- Utilizzare il comando `systemctl` per configurare il server Web Apache per l'avvio a ogni avvio del sistema.

```
[ec2-user ~]$ sudo systemctl enable httpd
```

Puoi verificare che `httpd` sia attivo eseguendo il seguente comando:

```
[ec2-user ~]$ sudo systemctl is-enabled httpd
```

- Se ancora non è stato fatto, aggiungere una regola di sicurezza per consentire le connessioni HTTP (porta 80) entranti all'istanza. Per impostazione predefinita, durante il lancio è stato creato un gruppo di **N** sicurezza `launch-wizard` per l'istanza. Se non hai aggiunto regole del gruppo di sicurezza supplementari, questo gruppo contiene una singola regola per consentire connessioni SSH.
 - Apri la console Amazon EC2 all'indirizzo <https://console.aws.amazon.com/ec2/>.
 - Nel riquadro di navigazione sinistro, scegli Istanze e seleziona la tua istanza.
 - Nella scheda Security (Sicurezza) visualizzare le regole in entrata. Verrà visualizzata la regola seguente:

Port range	Protocol	Source
22	tcp	0.0.0.0/0

Warning


L'utilizzo `0.0.0.0/0` consente a tutti gli IPv4 indirizzi di accedere all'istanza tramite SSH. L'opzione è accettabile per un breve periodo di tempo in un ambiente di test, ma non è sicura per gli ambienti di produzione. In produzione, potrai autorizzare solo un determinato indirizzo IP o un intervallo di indirizzi per accedere a un'istanza.

- Se non esiste alcuna regola in entrata per consentire connessioni HTTP (porta 80), la regola deve essere aggiunta ora. Scegliere il collegamento per il gruppo di sicurezza. Utilizzando le procedure descritte nella sezione [Autorizzazione del traffico in entrata per le istanze Linux](#), aggiungi una nuova regola di sicurezza in entrata con i seguenti valori:

- Type (Tipo): HTTP
 - Protocollo: TCP
 - Port Range (Intervallo porte): 80
 - Source (Origine): personalizzata
8. Verificare il server Web. Nel browser Web, digitare l'indirizzo DNS pubblico (o l'indirizzo IP pubblico) dell'istanza. In assenza di contenuti in `/var/www/html`, dovresti visualizzare la pagina di test di Apache che mostra il messaggio "Funziona!".

Puoi ottenere il DNS pubblico per la tua istanza utilizzando la console Amazon EC2 (controlla la colonna IPv4 Public DNS; se questa colonna è nascosta, scegli Preferenze (l'icona a forma di ingranaggio) e attiva Public DNS). IPv4

Verificare che il gruppo di sicurezza per l'istanza contenga una regola per consentire il traffico HTTP sulla porta 80. [Per ulteriori informazioni, consulta Aggiungere regole al gruppo di sicurezza.](#)

 Important

Se non utilizzi Amazon Linux, potrebbe inoltre essere necessario anche configurare il firewall sulla tua istanza per consentire queste connessioni. Per ulteriori informazioni sulla modalità di configurazione del firewall, consulta la documentazione per la distribuzione specifica.

Apache httpd utilizza i file che sono tenuti in una directory chiamata root del documento di Apache. La root del documento di Apache Amazon Linux è `/var/www/html`, che per impostazione predefinita è di proprietà della root.

Per permettere all'account `ec2-user` di manipolare file nella directory, è necessario modificare la proprietà e le autorizzazioni della directory. Sono disponibili molti modi per completare questa attività. In questo tutorial, aggiungi `ec2-user` al gruppo `apache` per assegnare la proprietà del gruppo `apache` della directory `/var/www` e assegnare autorizzazioni di scrittura al gruppo.

Per impostare le autorizzazioni dei file

1. Aggiungere l'utente (in questo caso `ec2-user`) al gruppo `apache`.

```
[ec2-user ~]$ sudo usermod -a -G apache ec2-user
```

2. Uscire e ripetere l'accesso per scegliere il nuovo gruppo, quindi verificare l'appartenenza.

a. Uscire (utilizzare il comando `exit` o chiudere la finestra terminale):

```
[ec2-user ~]$ exit
```

b. Per verificare l'appartenenza al gruppo `apache`, riconnettersi all'istanza, quindi eseguire il seguente comando:

```
[ec2-user ~]$ groups  
ec2-user adm wheel apache systemd-journal
```

3. Modificare la proprietà del gruppo di `/var/www` e dei suoi contenuti al gruppo `apache`.

```
[ec2-user ~]$ sudo chown -R ec2-user:apache /var/www
```

4. Per aggiungere le autorizzazioni di scrittura di gruppo e impostare l'ID di gruppo nelle sottodirectory future, modificare le autorizzazioni di directory di `/var/www` e delle relative sottodirectory.

```
[ec2-user ~]$ sudo chmod 2775 /var/www && find /var/www -type d -exec sudo chmod 2775 {} \;
```

5. Per aggiungere le autorizzazioni di scrittura di gruppo, modificare in modo ricorsivo le autorizzazioni del file di `/var/www` e delle relative sottodirectory:

```
[ec2-user ~]$ find /var/www -type f -exec sudo chmod 0664 {} \;
```

Ora, `ec2-user` (e qualsiasi membro futuro del gruppo `apache`) può aggiungere, eliminare e modificare i file nella root del documento di Apache, consentendoti di aggiungere contenuti, ad esempio un sito Web statico o un'applicazione PHP.

Per proteggere il server Web (facoltativo)

Un server Web che esegue il protocollo HTTP non offre alcuna sicurezza di trasporto per i dati inviati e ricevuti. Quando ti connetti a un server HTTP utilizzando un browser Web, URLs ciò che visiti, il contenuto delle pagine Web che ricevi e il contenuto (comprese le password) di tutti i moduli HTML

che invii sono tutti visibili agli intercettatori in qualsiasi punto del percorso di rete. La best practice per la protezione del tuo server Web prevede l'installazione del supporto per HTTPS (HTTP Secure), che protegge i dati con la crittografia SSL/TLS.

Per informazioni sull'abilitazione di HTTPS sul server, consulta [Tutorial: Configura SSL/TLS su AL2023](#).

Fase 2: verificare il server LAMP

Se il server è installato e in esecuzione e le autorizzazioni dei file sono impostate correttamente, l'account `ec2-user` dovrebbe essere in grado di creare un file PHP nella directory `/var/www/html` disponibile da Internet.

Per verificare il server LAMP

1. Creare un file PHP nella root del documento di Apache.


```
[ec2-user ~]$ echo "<?php phpinfo(); ?>" > /var/www/html/phpinfo.php
```

Se si verifica un errore "Permission denied" (Autorizzazione negata) quando si tenta di eseguire questo comando, provare a uscire e accedere nuovamente per ottenere le autorizzazioni di gruppo appropriate configurate in [Per impostare le autorizzazioni dei file](#).

2. In un browser Web, digitare l'URL del file appena creato. Questo URL è l'indirizzo DNS pubblico dell'istanza, seguito da una barra e dal nome di file. Ad esempio:


```
http://my.public.dns.amazonaws.com/phpinfo.php
```

Viene visualizzata la pagina delle informazioni PHP:

PHP Version 8.1.7


System	Linux ip-172-31-16-77.ec2.internal 5.15.57-28.127.amzn2022.aarch64 #1 SMP Thu Aug 4 17:06:57 UTC 2022 aarch64
Build Date	Jun 7 2022 18:21:38
Build System	Linux
Build Provider	Amazon Linux
Compiler	gcc (GCC) 11.3.1 20220421 (Red Hat 11.3.1-2)
Architecture	aarch64
Server API	FPM/FastCGI
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc
Loaded Configuration File	/etc/php.ini
Scan this dir for additional .ini files	/etc/php.d
Additional .ini files parsed	/etc/php.d/10-opcache.ini, /etc/php.d/20-bz2.ini, /etc/php.d/20-calendar.ini, /etc/php.d/20-ctype.ini, /etc/php.d/20-curl.ini, /etc/php.d/20-dom.ini, /etc/php.d/20-exif.ini, /etc/php.d/20-fileinfo.ini, /etc/php.d/20-ftp.ini, /etc/php.d/20-gd.ini, /etc/php.d/20-gettext.ini, /etc/php.d/20-iconv.ini, /etc/php.d/20-mbstring.ini, /etc/php.d/20-mysqlnd.ini, /etc/php.d/20-pdo.ini, /etc/php.d/20-phar.ini, /etc/php.d/20-simplexml.ini, /etc/php.d/20-sockets.ini, /etc/php.d/20-sqlite3.ini, /etc/php.d/20-tokenizer.ini, /etc/php.d/20-xml.ini, /etc/php.d/20-xmlwriter.ini, /etc/php.d/20-xsl.ini, /etc/php.d/30-mysqli.ini, /etc/php.d/30-pdo_mysql.ini, /etc/php.d/30-pdo_sqlite.ini, /etc/php.d/30-xmldrader.ini
PHP API	20210902
PHP Extension	20210902
Zend Extension	420210902
Zend Extension Build	API420210902,NTS
PHP Extension Build	API20210902,NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	enabled
Zend Memory Manager	enabled
Zend Multibyte Support	provided by mbstring
IPv6 Support	enabled
DTrace Support	available, disabled
Registered PHP Streams	https, ftps, compress.zlib, php, file, glob, data, http, ftp, compress.bzip2, phar
Registered Stream Socket Transports	tcp, udp, unix, udg, ssl, tls, tlsv1.0, tlsv1.1, tlsv1.2, tlsv1.3
Registered Stream Filters	zlib.*, string.rot13, string.toupper, string.tolower, convert.*, consumed, dechunk, bzip2.*, convert.iconv.*

This program makes use of the Zend Scripting Language Engine:
 Zend Engine v4.1.7, Copyright (c) Zend Technologies
 with Zend OPcache v8.1.7, Copyright (c), by Zend Technologies



Se non viene visualizzata questa pagina, verifica che il file `/var/www/html/phpinfo.php` sia stato creato correttamente nella fase precedente. È anche possibile verificare che tutti i pacchetti richiesti siano stati installati con il seguente comando.

```
[ec2-user ~]$ sudo dnf list installed httpd mariadb-server php-mysqlnd
```

Se uno dei pacchetti richiesti non è elencato nell'output, installarlo utilizzando il comando `sudo yum install package`.

3. Eliminare il file `phpinfo.php`. Sebbene questa informazione possa essere utile, non deve essere divulgata su Internet per ragioni di sicurezza.

```
[ec2-user ~]$ rm /var/www/html/phpinfo.php
```

Ora si dovrebbe avere un server Web LAMP completamente funzionante. Se vengono aggiunti contenuti alla root del documento di Apache su `/var/www/html`, dovrebbe essere possibile visualizzare tali contenuti all'indirizzo DNS pubblico per l'istanza.

Fase 3: proteggere il server di database

L'installazione predefinita del server MariaDB ha diverse caratteristiche che sono ottime per test e sviluppo, ma dovrebbero essere disabilitate o rimosse per i server di produzione. Il comando `mysql_secure_installation` guida attraverso il processo di impostazione di una password root e la rimozione delle caratteristiche non protette dall'installazione. Anche se non hai intenzione di utilizzare il server MariaDB, consigliamo di eseguire questa procedura.

Per proteggere il server MariaDB

1. Avviare il server MariaDB.

```
[ec2-user ~]$ sudo systemctl start mariadb
```

2. Esegui `mysql_secure_installation`.

```
[ec2-user ~]$ sudo mysql_secure_installation
```

- a. Quando richiesto, digitare una password per l'account root.
 - i. Digitare la password root corrente. Per impostazione predefinita, l'account root non ha una password configurata. Premere Invio.
 - ii. Digitare **Y** per impostare una password e digitare una password sicura due volte. Per ulteriori informazioni sulla creazione di una password sicura, vedere <https://identitysafe.norton.com/password-generator/>. Assicurarsi di conservare questa password in un posto sicuro.

L'impostazione di una password root per MariaDB è solo la misura di base per la protezione del database. Quando si crea o si installa un'applicazione basata su un

database, normalmente si crea un utente del servizio di database per tale applicazione per evitare di usare l'account root per ragioni diverse dall'amministrazione del database.

- b. Digitare **Y** per rimuovere gli account utente anonimi.
 - c. Digitare **Y** per disabilitare l'accesso root in remoto.
 - d. Digitare **Y** per rimuovere il database di test.
 - e. Digitare **Y** per ricaricare le tabelle dei privilegi e salvare le modifiche.
3. (Facoltativo) Se non si ha intenzione di utilizzare immediatamente il server MariaDB, interromperlo. È possibile riavviarlo quando è di nuovo necessario.

```
[ec2-user ~]$ sudo systemctl stop mariadb
```

4. (Facoltativo) Se si desidera che il server MariaDB si avvii a ogni avvio, digitare il seguente comando.

```
[ec2-user ~]$ sudo systemctl enable mariadb
```

Fase 4: Installazione (facoltativa) phpMyAdmin

[phpMyAdmin](#) è uno strumento di gestione dei database basato sul Web che puoi utilizzare per visualizzare e modificare i database MySQL sulla tua istanza EC2. Segui le fasi seguenti per installare e configurare phpMyAdmin sull'istanza Amazon Linux.

Important

Non è consigliabile phpMyAdmin utilizzarlo per accedere a un server LAMP a meno che tu non lo abbia abilitato SSL/TLS in Apache; in caso contrario, la password dell'amministratore del database e altri dati vengono trasmessi in modo non sicuro su Internet. Per i consigli sulla sicurezza forniti dagli sviluppatori, consulta [Proteggere](#) l'installazione. phpMyAdmin Per informazioni generali sulla protezione di un server Web su un'istanza EC2, consulta [Tutorial: Configura SSL/TLS su AL2023](#).

Per installare phpMyAdmin

1. Installare le dipendenze richieste.

```
[ec2-user ~]$ sudo dnf install php-mbstring php-xml -y
```

2. Riavviare Apache.

```
[ec2-user ~]$ sudo systemctl restart httpd
```

3. Riavviare php-fpm.

```
[ec2-user ~]$ sudo systemctl restart php-fpm
```

4. Andare alla root del documento di Apache in `/var/www/html`.

```
[ec2-user ~]$ cd /var/www/html
```

5. Seleziona un pacchetto sorgente per l'ultima phpMyAdmin versione da <https://www.phpmyadmin.net/downloads>. Per scaricare il file direttamente nell'istanza, copiare il link e incollarlo in un comando wget, come in questo esempio:

```
[ec2-user html]$ wget https://www.phpmyadmin.net/downloads/phpMyAdmin-latest-all-languages.tar.gz
```

6. Creare una cartella phpMyAdmin in cui estrarre il pacchetto con il comando seguente.

```
[ec2-user html]$ mkdir phpMyAdmin && tar -xvzf phpMyAdmin-latest-all-languages.tar.gz -C phpMyAdmin --strip-components 1
```

7. Eliminare il `phpMyAdmin-latest-all-languages.tar.gz` tarball.

```
[ec2-user html]$ rm phpMyAdmin-latest-all-languages.tar.gz
```

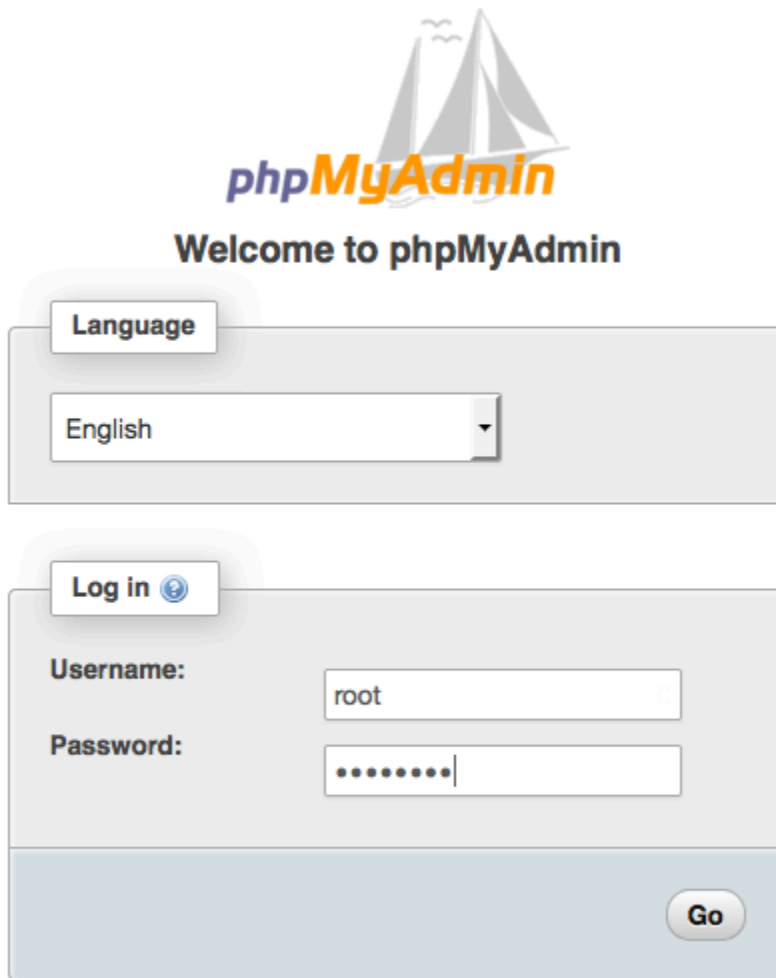
8. (Facoltativo) Se il server MySQL non è in esecuzione, avviarlo in questo momento.

```
[ec2-user ~]$ sudo systemctl start mariadb
```

9. In un browser Web, digita l'URL dell' phpMyAdmin installazione. Questo URL è l'indirizzo DNS pubblico (o indirizzo IP pubblico) dell'istanza seguito da una barra e dal nome della directory di installazione. Esempio:

```
http://my.public.dns.amazonaws.com/phpMyAdmin
```

Dovresti vedere la pagina phpMyAdmin di accesso:



The image shows the phpMyAdmin login interface. At the top, there is a logo with a sailboat and the text 'phpMyAdmin'. Below the logo, it says 'Welcome to phpMyAdmin'. There are two main sections: a 'Language' section with a dropdown menu currently set to 'English', and a 'Log in' section. The 'Log in' section contains a 'Username:' field with 'root' entered, a 'Password:' field with masked characters (dots), and a 'Go' button at the bottom right.

10. Accedi all' phpMyAdmin installazione con il nome `root` utente e la password `root` MySQL che hai creato in precedenza.

L'installazione deve essere configurata prima di essere messa in funzione. Si consiglia di iniziare con la creazione manuale del file di configurazione, come segue:

- a. Per iniziare con un file di configurazione minimo, utilizza l'editor di testo preferito per creare un nuovo file e quindi copia al suo interno il contenuto di `config.sample.inc.php`.
- b. Salva il file come `config.inc.php` nella phpMyAdmin directory che contiene `index.php`
- c. Per qualsiasi [configurazione aggiuntiva, fate riferimento alle istruzioni successive alla creazione del file nella sezione Uso dello script](#) di phpMyAdmin installazione delle istruzioni di installazione.

Per informazioni sull'utilizzo phpMyAdmin, consultate la [Guida per l'phpMyAdmin utente](#).

Risoluzione dei problemi

Questa sezione offre suggerimenti per la risoluzione di problemi comuni che possono verificarsi durante la configurazione di un nuovo server LAMP.

Non riesco a connettermi al mio server utilizzando un browser Web

Esegui i controlli seguenti per verificare se il tuo server Web Apache è in esecuzione e accessibile.

- Il server Web è in esecuzione?

Puoi verificare che httpd sia attivo eseguendo il seguente comando:

```
[ec2-user ~]$ sudo systemctl is-enabled httpd
```

Se il processo httpd non è in esecuzione, ripeti le fasi descritte in [Per preparare il server LAMP](#).

- Il firewall è configurato correttamente?

Verificare che il gruppo di sicurezza per l'istanza contenga una regola per consentire il traffico HTTP sulla porta 80. Per ulteriori informazioni, consulta [Aggiungere regole al gruppo di sicurezza](#).

Non riesco a connettermi al mio server utilizzando HTTPS

Eseguire le seguenti verifiche per verificare se il server Web Apache è configurato per supportare HTTPS.

- Il server Web è configurato correttamente?

Dopo avere installato Apache, il server è configurato per il traffico HTTP. Per supportare HTTPS, abilitare TLS sul server e installare un certificato SSL. Per informazioni, consulta [Tutorial: Configura SSL/TLS su AL2023](#).

- Il firewall è configurato correttamente?

Verificare che il gruppo di protezione per l'istanza contenga una regola per consentire il traffico HTTPS sulla porta 443. Per ulteriori informazioni, consulta [Autorizzare il traffico in entrata per le istanze Linux](#).

Argomenti correlati

Per ulteriori informazioni sul trasferimento di file sull'istanza o sull'installazione di un WordPress blog sul server Web, consulta la seguente documentazione:

- [Trasferisci file sulla tua istanza Linux utilizzando WinSCP](#) nella Amazon EC2 User Guide.
- [Trasferisci file su istanze Linux utilizzando un client SCP](#) nella Amazon EC2 User Guide.
- [Tutorial: Ospita un WordPress blog su AL2023](#)

Per ulteriori informazioni sui comandi e sul software utilizzati in questo tutorial, consulta le pagine Web seguenti:

- Server Web Apache: <http://httpd.apache.org/>
- Server database MariaDB: <https://mariadb.org/>
- Linguaggi di programmazione PHP: <http://php.net/>

Per ulteriori informazioni sulla registrazione di un nome di dominio per il server Web o sul trasferimento di un nome di dominio esistente su questo host, consulta l'articolo relativo alla [creazione e alla migrazione di domini e sottodomini ad Amazon Route 53](#) nella Guida per lo sviluppatore di Amazon Route 53.

Tutorial: Configura SSL/TLS su AL2023

Secure Layer/Transport Sockets Layer Security (SSL/TLS) creates an encrypted channel between a web server and web client that protects data in transit from being eavesdropped on. This tutorial explains how to add support manually for SSL/TLS su un'istanza EC2 con AL2023 un server web Apache). Questo tutorial presuppone che si non stia utilizzando un sistema di bilanciamento del carico (load balancer). Se si utilizza Elastic Load Balancing, è possibile scegliere di configurare l'offload SSL sul load balancer, utilizzando invece un certificato di [AWS Certificate Manager](#).

Per motivi storici, la crittografia Web viene spesso definita semplicemente con l'acronimo SSL. Se da un lato i browser Web continuano a supportare il protocollo SSL, dall'altro il protocollo TLS, suo successore, è meno vulnerabile agli attacchi. Per impostazione predefinita, AL2023 disabilita il supporto lato server per tutte le versioni di SSL. Gli [organismi che si occupano degli standard di sicurezza](#) considerano TLS 1.0 non sicuro. TLS 1.0 e TLS 1.1 sono stati dichiarati formalmente [obsoleti](#) a marzo 2021. Le istruzioni contenute in questo tutorial si basano esclusivamente sull'abilitazione di TLS 1.2. TLS 1.3 è stato finalizzato nel 2018 ed è disponibile AL2 purché la libreria TLS sottostante (OpenSSL in questo tutorial) sia supportata e abilitata. [I clienti devono supportare TLS 1.2 o versioni successive entro il 28 giugno 2023](#). Per ulteriori informazioni sugli standard di crittografia aggiornati, consulta [RFC 7568](#) e [RFC 8446](#).

Questo tutorial fa riferimento alla crittografia Web moderna semplicemente come TLS.

Important

Queste procedure sono destinate all'uso con AL2023. Se si sta tentando di impostare un'istanza EC2 che esegue una distribuzione diversa o un'istanza che esegue una versione precedente di Amazon Linux 2, alcune procedure in questa esercitazione potrebbero non funzionare. Per Ubuntu, consulta la documentazione seguente della community Ubuntu: [Open SSL on Ubuntu](#). Per Red Hat Enterprise Linux, consulta il seguente argomento: [Setting up the Apache HTTP Web Server](#) (Configurazione del server Web HTTP Apache). Per altre distribuzioni, consulta la relativa documentazione specifica.

Note

In alternativa, puoi utilizzare AWS Certificate Manager (ACM) for AWS Nitro enclaves, un'applicazione enclave che consente di utilizzare SSL/TLS certificati pubblici e privati con applicazioni Web e server in esecuzione su istanze Amazon EC2 con Nitro Enclaves. AWS Nitro Enclaves è una funzionalità di Amazon EC2 che consente la creazione di ambienti di elaborazione isolati per proteggere ed elaborare in modo sicuro dati altamente sensibili, come certificati e chiavi private. SSL/TLS ACM per Nitro Enclaves funziona con nginx in esecuzione sull'istanza Amazon EC2 Linux per creare chiavi private, distribuire certificati e chiavi private e gestire i rinnovi dei certificati. Per utilizzare ACM per Nitro Enclaves, è necessario utilizzare un'istanza Linux abilitata all'enclave.

[Per ulteriori informazioni, consulta *Che cos'è Nitro Enclaves? AWS* e *AWS Certificate Manager per Nitro Enclaves nella Guida per l'utente di Nitro Enclaves*.AWS](#)

Indice

- [Prerequisiti](#)
- [Fase 1: abilitare TLS nel server](#)
- [Fase 2: ottenere un certificato firmato dalla CA](#)
- [Fase 3: testare e proteggere la configurazione di sicurezza](#)
- [Risoluzione dei problemi](#)

Prerequisiti

Prima di iniziare questo tutorial, completare le procedure descritte di seguito:

- Avvia un' AL2023 istanza supportata da EBS. Per ulteriori informazioni, consulta [AL2023 su Amazon EC2](#).
- Configurare i gruppi di sicurezza in modo da consentire all'istanza di accettare le connessioni sulle porte TCP seguenti:
 - SSH (porta 22)
 - HTTP (porta 80)
 - HTTPS (porta 443)

Per ulteriori informazioni, consulta [Autorizza il traffico in entrata per le tue istanze Linux](#) nella Guida per l'utente di Amazon EC2.

- Installare il server Web Apache. Per step-by-step istruzioni, consulta. [Tutorial: Installare un server LAMP su AL2023](#) Sono necessari solo il pacchetto httpd e le relative dipendenze. Puoi pertanto ignorare le istruzioni relative a PHP e MariaDB.
- Per identificare e autenticare i siti Web, l'infrastruttura a chiave pubblica (PKI) TLS si basa su Domain Name System (DNS). Per utilizzare l'istanza EC2 per ospitare un sito Web pubblico, devi registrare un nome di dominio per il server Web o trasferire un nome di dominio esistente nell'host Amazon EC2. Per questa operazione sono disponibili numerosi servizi di registrazione di domini e hosting DNS di terze parti. In alternativa, puoi utilizzare [Amazon Route 53](#).

Fase 1: abilitare TLS nel server

Questa procedura illustra il processo di configurazione di TLS AL2023 con un certificato digitale autofirmato.

Note

Un certificato autofirmato è accettabile in ambienti di test, ma non in ambienti di produzione. Se esponi un certificato autofirmato in Internet, i visitatori del sito visualizzeranno avvisi di sicurezza.

Per abilitare TLS in un server

1. Connettersi all'istanza e confermare che Apache è in esecuzione. Per ulteriori informazioni, consulta [AL2023 Connessione alle istanze](#).

```
[ec2-user ~]$ sudo systemctl is-enabled httpd
```

Se il valore restituito non è "enabled" ("abilitato"), avviare Apache e configurarlo in modo che venga avviato all'avvio del sistema:

```
[ec2-user ~]$ sudo systemctl start httpd && sudo systemctl enable httpd
```

2. Per verificare che tutti i pacchetti software siano aggiornati, eseguire un aggiornamento rapido del software sull'istanza. Questo processo può richiedere alcuni minuti, ma è importante assicurarsi di disporre della versione più recente degli aggiornamenti della sicurezza e delle correzioni dei bug.

Note

L'opzione `-y` installa gli aggiornamenti senza chiedere conferma. Se desideri esaminare gli aggiornamenti prima di installarli, puoi omettere questa opzione.

```
[ec2-user ~]$ sudo dnf install openssl mod_ssl
```

3. Dopo aver inserito il seguente comando, verrà indirizzato a un prompt in cui è possibile immettere le informazioni sul sito.

```
[ec2-user ~]$ sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/pki/tls/private/apache-selfsigned.key -out /etc/pki/tls/certs/apache-selfsigned.crt
```

Questo genera due file:

- `/etc/pki/tls/private/apache-selfsigned.key`(la chiave privata)
- `/etc/pki/tls/certs/apache-selfsigned.crt`(il certificato autofirmato)

Il nome del file del certificato corrisponde all'impostazione predefinita assegnata nella `SSLCertificateFile` direttiva in `/etc/httpd/conf.d/ssl.conf`.

L'istanza dispone ora dei file seguenti, che serviranno per configurare il server sicuro e creare un certificato per il test:

- `/etc/httpd/conf.d/ssl.conf`

File di configurazione per `mod_ssl`. Contiene le direttive che indicano ad Apache dove cercare le chiavi e i certificati di crittografia, le versioni del protocollo TLS da consentire e il tipo di crittografia da accettare. Questo sarà il file di certificato locale:

- `/etc/pki/tls/certs/apache-selfsigned.crt`
- `/etc/pki/tls/private/apache-selfsigned.key`

Il `.crt` file contiene il certificato autofirmato e il `.key` file contiene la chiave privata. Apache richiede che certificato e chiave siano entrambi in formato PEM, che è composto da caratteri ASCII con codifica Base64 racchiusi tra le righe "BEGIN" ed "END", come nell'esempio abbreviato riportato di seguito.

```
-----BEGIN PRIVATE KEY-----
MIIEvgIBADANBgkqhkiG9w0BAQEFAASCBAgwgwggSkAgEAAoIBAQD2KKx/8Zk94m1q
3gQMZF9ZN66Ls19+3tHAgQ5Fpo9KJDhzLj00CI8u1PTcGmAah5kEitCEc0wzmNeo
BC10wYR6G0rGaKtK9Dn7CuIjvubtUysVyQoMVPQ97ldeakHWeRMiEJFXg6kZZ0vr
GvwnKoMh3D1K44D9dX7IDua2P1Yx5+eroA+1Lqf32ZSaA00bBIMIYTHigwbHMZoT
...
56tE7THvH7v0Ef4/iU0sIrEzaMaJ0mqkmY1A70qQGQKBgBF3H1qNRNHuyMcPODFs
```

```

27hDzPDinrquSEvoZIggkDM1h2irTiipJ/GhkvTpoQ1v0fK/VXw8vSgeaBuhwJvS
LXU9HvYq0U604FgD3nAyB9hI0BE13r1HjUvbjT7moH+RhnNz6eqqscCS09VtRAo
4QQvAq0a8UheYeoXLdWcHaLP
-----END PRIVATE KEY-----

-----BEGIN CERTIFICATE-----
MIIEazCCA10gAwIBAgICWxQwDQYJKoZIhvcNAQELBQAwbExCzAJBgNVBAYTAi0t
MRIwEAYDVQQIDAlTb211U3RhdGUxETAPBgNVBACMFNvbWVwVWVWVWVWVWVWVW
DBBTb211T3JnYW5pemF0aW9uMR8wHQYDVQQLDBZTb211T3JnYW5pemF0aW9uYXV
bm10MRkwFwYDVQQDDDBpcC0xNzItMzEtMjAtMjMMSQwIgwYJKoZIhvcNAQkBFhVy
...
z5rRUE/XzxRLBZ0oWZpNWTXJkQ3uFYH6s/sBwtHpKKZMz0vDedREjNKAvk4ws6F0
CuIjvubtUysVYqoMVPQ971deakHWeRMiEJFXg6kZZ0vrGvwnKoMh3D1K44D9d1U3
WanXWehT6FiSZvB4sTEXXJN2jdw8g+sHGnZ8zC0sc1knYhHrCVD2vnB1ZJKSZvak
3ZazhBxtQSukFM0nWPP2a0DMMFGYUH0d0BQE8sBJxg==
-----END CERTIFICATE-----

```

I nomi e le estensioni di file rappresentano una convenzione e non hanno alcuna ripercussione sulla funzionalità. Ad esempio è possibile denominare un certificato `cert.crt`, `cert.pem` o con qualsiasi altro nome di file, a condizione che la direttiva corrispondente nel file `ssl.conf` utilizzi lo stesso nome.

Note

Quando si sostituiscono i file TLS predefiniti con file personalizzati, assicurarsi che siano in formato PEM.

4. Riavviare Apache.

```
[ec2-user ~]$ sudo systemctl restart httpd
```

Note

Assicurarsi che la porta TCP 443 sia accessibile sull'istanza EC2, come descritto in precedenza.

5. Il server Web Apache ora dovrebbe supportare HTTPS (HTTP protetto) sulla porta 443. Per eseguire il test, digitare l'indirizzo IP o il nome di dominio completo dell'istanza EC2 nella barra degli indirizzi URL di un browser con il prefisso **https://**.

Poiché ti stai connettendo a un sito con un certificato host autofirmato non attendibile, il browser potrebbe visualizzare una serie di avvisi di sicurezza. Ignorare gli avvisi e passare al sito.

Se la pagina predefinita di test di Apache viene visualizzata, significa che TLS è stato correttamente configurato sul server. Tutti i dati in transito tra il browser e il server ora sono crittografati.

Note

Per evitare che i visitatori del sito vedano schermate di avviso, è necessario ottenere un certificato attendibile che non solo esegua la crittografia, ma che fornisca anche un'autenticazione pubblica del proprietario del sito.

Fase 2: ottenere un certificato firmato dalla CA

Puoi utilizzare la seguente procedura per ottenere un certificato firmato dalla CA:

- Generare una richiesta di firma del certificato (CSR) da una chiave privata
- Inviare il CSR alla Certificate Authority
- Ottenere un certificato host firmato
- Configurare Apache per utilizzare il certificato


Dal punto di vista della crittografia un certificato host TLS X.509 autofirmato è identico a un certificato firmato da una CA. La differenza è una questione di attendibilità. Una CA si impegna infatti a fornire una convalida minima della titolarità di un dominio prima di emettere un certificato a un richiedente. Ogni browser Web contiene un elenco di quelle CAs ritenute affidabili dal fornitore del browser a tale scopo. Un certificato X.509 è principalmente composto da una chiave server privata e da una firma fornita dalla CA e associata a livello di crittografia alla chiave pubblica. Quando un browser si connette a un server Web tramite HTTPS, il server presenta un certificato da confrontare con l'elenco dei siti attendibili CAs. Se il firmatario è incluso nell'elenco oppure è accessibile tramite una catena di attendibilità composta da altri firmatari fidati, il browser negozia un canale di dati a crittografia rapida con il server e carica la pagina.

I certificati in genere costano poiché il processo di convalida delle richieste prevede alcuni costi. Consigliamo pertanto di valutare le varie offerte. Alcuni CAs offrono certificati di livello base gratuiti.

Il più importante di questi CAs è il progetto [Let's Encrypt](#), che supporta anche l'automazione del processo di creazione e rinnovo dei certificati. Per ulteriori informazioni sull'utilizzo di un certificato Let's Encrypt, consulta la pagina [Ottenimento di Certbot](#).

Se hai intenzione di offrire servizi di livello commerciale, [AWS Certificate Manager](#) è una buona opzione.

L'uso di un certificato host sottostante rappresenta la soluzione ideale. Dal 2019, gruppi appartenenti alla [pubblica amministrazione](#) e a [settori](#) specifici consigliano una dimensione (modulo) di chiave minima pari a 2048 bit per le chiavi RSA a protezione dei documenti fino al 2030. La dimensione predefinita del modulo generato da OpenSSL AL2023 in è di 2048 bit, adatta per l'uso in un certificato firmato da un'autorità di certificazione. Nella seguente procedura viene offerto un passaggio opzionale per coloro che desiderano una chiave personalizzata, ad esempio, una chiave con un modulo più grande o che utilizza un algoritmo di crittografia diverso.

 Important

In mancanza di un dominio DNS registrato e ospitato, tali istruzioni per l'acquisizione di certificati host firmati dalla CA non funzioneranno.

Per ottenere un certificato firmato dalla CA

1. Connect alla propria istanza e accedi a `/etc/pki/tls/private/`. Si tratta della directory in cui viene memorizzata la chiave privata del server per TLS. Se preferisci utilizzare una chiave host esistente per generare la CSR, passa alla Fase 3. Per ulteriori informazioni sulla connessione alla tua istanza, vedi [AL2023 Connessione alle istanze](#)
2. (Opzionale) Generare una nuova chiave privata. Di seguito sono riportate alcune configurazioni di chiave di esempio. Qualsiasi chiave risultante funziona con il server Web, ma il livello e il tipo di sicurezza implementati possono variare.
 - Esempio 1: creare una chiave host RSA predefinita. Il file risultante, **custom.key**, è una chiave privata RSA a 2048 bit.

```
[ec2-user ~]$ sudo openssl genrsa -out custom.key
```

- Esempio 2: creare una chiave RSA più complessa con un modulo più grande, Il file risultante, **custom.key**, è una chiave privata RSA a 4096 bit.

```
[ec2-user ~]$ sudo openssl genrsa -out custom.key 4096
```

- Esempio 3: creare una chiave RSA crittografata a 4096 bit con protezione con password. Il file risultante, **custom.key**, è una chiave privata RSA a 4096 bit crittografata in base allo standard AES-128.

Important

La crittografia di una chiave fornisce maggiore sicurezza, ma dal momento che una chiave crittografata richiede una password, i servizi che dipendono da essa non possono essere avviati automaticamente. Ogni volta che usi questa chiave, devi fornire la password (nell'esempio precedente, "abcde12345") tramite una connessione SSH.

```
[ec2-user ~]$ sudo openssl genrsa -aes128 -passout pass:abcde12345 -out custom.key 4096
```

- Esempio 4: creare una chiave utilizzando uno standard non RSA. La crittografia RSA può essere relativamente lenta per via della dimensione delle chiavi pubbliche, che sono basate sul prodotto di due grandi numeri primi. Tuttavia, è possibile creare chiavi per TLS che utilizzano una crittografia non RSA. Le chiavi basate su calcoli matematici di curve ellittiche sono di dimensioni inferiori e, dal punto di vista del calcolo, più rapide pur garantendo un livello equivalente di sicurezza.

```
[ec2-user ~]$ sudo openssl ecparam -name prime256v1 -out custom.key -genkey
```

Il risultato è una chiave privata basata su curva ellittica a 256 bit che utilizza prime256v1, una "curva denominata" supportata da OpenSSL. La complessità dal punto di vista crittografico è leggermente superiore rispetto una chiave RSA a 2048 bit, [secondo i dati NIST](#).

Note

Non tutte CAs forniscono lo stesso livello di supporto per elliptic-curve-based le chiavi RSA.

Assicurati che la nuova chiave privata abbia proprietà e autorizzazioni estremamente restrittive (owner=root, group=root, solo per il proprietario). read/write Il comando è come mostrato nell'esempio seguente.

```
[ec2-user ~]$ sudo chown root:root custom.key
[ec2-user ~]$ sudo chmod 600 custom.key
[ec2-user ~]$ ls -al custom.key
```

I comandi precedenti restituiscono il seguente risultato:

```
-rw----- root root custom.key
```

Dopo aver creato e configurato una chiave affidabile, puoi creare una CSR.

3. Creare una CSR utilizzando la chiave preferita. Nell'esempio seguente viene utilizzato **custom.key**.

```
[ec2-user ~]$ sudo openssl req -new -key custom.key -out csr.pem
```

OpenSSL visualizza una finestra di dialogo e richiede l'immissione delle informazioni riportate nella seguente tabella. Tutti i campi, tranne Common Name (Nome comune), sono facoltativi per un certificato host di base convalidato a livello di dominio.

Nome	Descrizione	Esempio
Nome paese	L'abbreviazione ISO di due lettere per il tuo paese.	US = Stati Uniti
State or Province Name (Nome stato o provincia)	Il nome dello stato o della provincia in cui si trova la tua organizzazione. Questo nome non può essere abbreviato.	Washington
Locality Name	La località in cui si trova la tua organizzazione, ad esempio una città.	Seattle

Nome	Descrizione	Esempio
(Nome località)		
Nome organizzazione	La denominazione legale completa della tua organizzazione. Non abbreviare il nome dell'organizzazione.	Example Corporation
Organizational Unit Name (Nome unità organizzativa)	Eventuali informazioni aggiuntive.	Example Dept
Common Name (Nome comune)	Questo valore deve corrispondere esattamente all'indirizzo Web che presumibilmente gli utenti immettono in un browser. In genere, ciò significa un nome di dominio con un nome host o alias con un prefisso, nel formato www.example.com . Nei test con un certificato autofirmato e senza risoluzione DNS, il nome comune può essere costituito solo dal nome host. CAs offrono anche certificati più costosi che accettano nomi wild-card come. *.example.com	www.example.com
Indirizzo e-mail	L'indirizzo e-mail dell'amministratore del server.	someone@example.com

Infine, OpenSSL richiede l'immissione di una password di verifica opzionale. Questa password è valida solo per la CSR e per le transazioni tra te e la CA. Pertanto, attieniti alle raccomandazioni della CA in merito alla definizione di questo tipo di password e all'altro campo facoltativo, ovvero il nome azienda facoltativo. La password di verifica associata alla CSR non ha alcuna ripercussione sulla funzionalità del server.

Il file **csr.pem** risultante contiene la chiave pubblica, la firma digitale della chiave pubblica e i metadati immessi.

- Inviare la CSR a una CA. In genere, questa operazione prevede l'apertura del file CSR in un editor di testo e la copia del contenuto in un modulo Web. In questo momento, è possibile che ti venga chiesto di fornire uno o più nomi alternativi del soggetto (SANs) da inserire nel certificato. Se **www.example.com** è il nome comune, **example.com** potrebbe essere un nome alternativo di oggetto (SAN) valido e viceversa. Un visitatore del sito che immettesse uno di questi due nomi avrebbe accesso a una connessione priva di errori. Se il modulo web CA lo consente, includi il nome comune nell'elenco di SANs. Alcuni lo CAs includono automaticamente.

Dopo l'approvazione della richiesta, riceverai un nuovo certificato host firmato dalla CA. Ti potrebbe inoltre venire richiesto di scaricare un file di certificato intermedio contenente i certificati aggiuntivi necessari per completare la catena di attendibilità della CA.

Note

La CA potrebbe inviare i file in più formati, destinati a scopi specifici. Ai fini di questo tutorial, ti consigliamo di usare solo un file di certificato in formato PEM, che in genere, ma non sempre, è contrassegnato dall'estensione `.pem` o `.crt`. Se non sei sicuro di quale file usare, apri il file in un editor di testo e cerca quello contenente uno o più blocchi che iniziano con la seguente riga.

```
- - - - -BEGIN CERTIFICATE - - - - -
```

Il file deve inoltre terminare con la seguente riga.

```
- - - - -END CERTIFICATE - - - - -
```

Puoi anche testare il file nella riga di comando come indicato di seguito.

```
[ec2-user certs]$ openssl x509 -in certificate.crt -text
```

Verifica che nel file appaiano queste righe. Non utilizzare file che terminano con `.p7b`, `.p7c` o estensioni simili.

- Posizionare il nuovo certificato firmato dalla CA ed eventuali certificati intermedi nella directory `/etc/pki/tls/certs`.

Note

Esistono vari modi per caricare il nuovo certificato nell'istanza EC2, ma il più semplice e immediato prevede di aprire un editor di testo (ad esempio, vi, nano o notepad) sul computer locale e sull'istanza e quindi di copiare e incollare il contenuto del file in queste posizioni. Devi disporre delle autorizzazioni root [sudo] durante l'esecuzione di queste operazioni nell'istanza EC2. In questo modo, puoi verificare in tempo reale se si verificano problemi a livello di autorizzazioni o percorsi. Presta particolare attenzione a non aggiungere altre righe durante la copia del contenuto o a non apportare modifiche di alcun tipo.

Dall'interno della `/etc/pki/tls/certs` directory, verifica che le impostazioni di proprietà del file, gruppo e autorizzazione corrispondano ai AL2023 valori predefiniti altamente restrittivi (owner=root, group=root, solo per il proprietario). read/write L'esempio seguente mostra i comandi da utilizzare.

```
[ec2-user certs]$ sudo chown root:root custom.crt
[ec2-user certs]$ sudo chmod 600 custom.crt
[ec2-user certs]$ ls -al custom.crt
```

Questi comandi dovrebbero restituire il seguente risultato.

```
-rw----- root root custom.crt
```


Le autorizzazioni del file del certificato intermedio sono meno rigide (owner=root, group=root, il proprietario può scrivere, il gruppo può leggere, tutti gli utenti possono leggere). L'esempio seguente mostra i comandi da utilizzare.

```
[ec2-user certs]$ sudo chown root:root intermediate.crt
[ec2-user certs]$ sudo chmod 644 intermediate.crt
[ec2-user certs]$ ls -al intermediate.crt
```

Questi comandi dovrebbero restituire il seguente risultato.

```
-rw-r--r-- root root intermediate.crt
```

6. Posizionare la chiave privata utilizzata per creare la CRS nella directory `/etc/pki/tls/private/`.

 Note

Esistono vari modi per caricare la chiave personalizzata nell'istanza EC2, ma il più semplice e immediato prevede di aprire un editor di testo (ad esempio, vi, nano o notepad) sul computer locale e sull'istanza e quindi di copiare e incollare il contenuto del file in queste posizioni. Devi disporre delle autorizzazioni root [sudo] durante l'esecuzione di queste operazioni nell'istanza EC2. In questo modo, puoi verificare in tempo reale se si verificano problemi a livello di autorizzazioni o percorsi. Presta particolare attenzione a non aggiungere altre righe durante la copia del contenuto o a non apportare modifiche di alcun tipo.


Dall'interno della `/etc/pki/tls/private` directory, usa i seguenti comandi per verificare che le impostazioni di proprietà, gruppo e autorizzazione dei file corrispondano ai valori AL2023 predefiniti altamente restrittivi (`owner=root, group=root, solo per il proprietario`). `read/write`

```
[ec2-user private]$ sudo chown root:root custom.key
[ec2-user private]$ sudo chmod 600 custom.key
[ec2-user private]$ ls -al custom.key
```

Questi comandi dovrebbero restituire il seguente risultato.

```
-rw----- root root custom.key
```

7. Modificare `/etc/httpd/conf.d/ssl.conf` per riflettere i nuovi file del certificato e della chiave.

 Note

Assicurati che sia impostato nel tuo blocco. `SSLEngine` on `VirtualHost`

- a. Indicare il percorso e il nome del file del certificato host firmato dalla CA nella direttiva `SSLCertificateFile` di Apache:

```
SSLCertificateFile /etc/pki/tls/certs/custom.crt
```

- b. In caso di ricezione di un file del certificato intermedio (`intermediate.crt` in questo esempio), specificare il relativo percorso e nome di file utilizzando la direttiva `SSLCACertificateFile` di Apache:

```
SSLCACertificateFile /etc/pki/tls/certs/intermediate.crt
```

Note

Alcuni CAs combinano il certificato host e i certificati intermedi in un unico file, rendendo superflua la `SSLCACertificateFile` direttiva. Consultare le istruzioni fornite dalla CA.

- c. Specificare il percorso e il nome del file della chiave privata (`custom.key` in questo esempio) nella direttiva `SSLCertificateKeyFile` di Apache:

```
SSLCertificateKeyFile /etc/pki/tls/private/custom.key
```

8. Salvare `/etc/httpd/conf.d/ssl.conf` e riavviare Apache.

```
[ec2-user ~]$ sudo systemctl restart httpd
```

9. Testare il server digitando il nome del dominio nella barra dell'URL di un browser con il prefisso `https://`. Il browser deve caricare la pagina di test su HTTPS senza errori.

Fase 3: testare e proteggere la configurazione di sicurezza

Dopo aver configurato TLS e averlo esposto al pubblico, devi testarne il livello effettivo di sicurezza. Questa operazione è semplice grazie a servizi online quali [Qualys SSL Labs](#), che eseguono un'analisi completa e gratuita della configurazione della sicurezza. In base ai risultati, puoi decidere di rafforzare la configurazione di sicurezza di default mediante il controllo dei protocolli accettati, del tipo di cifratura preferito e degli elementi da escludere. Per ulteriori informazioni, consulta la sezione relativa alla [formulazione delle classificazioni di Qualys](#).

⚠ Important

Il test in un ambiente reale è di cruciale importanza per la sicurezza del server. Piccoli errori di configurazione potrebbero generare gravi violazioni della sicurezza e perdita di dati. Poiché le procedure consigliate per la sicurezza sono in costante cambiamento in risposta a programmi di ricerca e minacce emergenti, verifiche periodiche della sicurezza rappresentano una pratica di amministrazione ottimale dei server.

Nel sito [Qualys SSL Labs](#), immetti il nome di dominio completo del server nel formato **www.example.com**. Dopo circa due minuti riceverai una valutazione del sito (da A a F) e un'analisi dettagliata dei risultati. La tabella seguente riassume il rapporto per un dominio con impostazioni identiche alla configurazione predefinita di Apache e con un certificato Certbot predefinito. AL2023

Valutazione complessiva	B
Certificato	100%
Supporto dei protocolli	95%
Scambio di chiavi	70%
Affidabilità crittografia	90%

Benché dalla panoramica emerga una certa solidità della configurazione, il rapporto dettagliato mette in luce diversi potenziali problemi, qui elencati in ordine di gravità:

X Il RC4 codice è supportato per l'uso da parte di alcuni browser meno recenti. Un codice è il nucleo matematico di un algoritmo di crittografia. RC4, [un codice veloce utilizzato per crittografare i flussi di dati TLS, è noto per presentare diversi gravi punti deboli](#). A meno di avere ottime ragioni per supportare browser legacy, è necessario disabilitare questa opzione.

X Sono supportate versioni di TLS meno recenti. La configurazione supporta TLS 1.0 (già obsoleto) e TLS 1.1 (in procinto di diventare obsoleto). A partire dal 2018, è raccomandato soltanto TLS 1.2.

X La proprietà Forward Secrecy non è completamente supportata. La proprietà [Forward Secrecy](#) è una caratteristica degli algoritmi che eseguono la crittografia utilizzando chiavi di sessione temporanee (effimere) derivate dalla chiave privata. Ciò in pratica significa che gli utenti

malintenzionati non possono decriptare i dati HTTPS anche se sono in possesso della chiave privata a lungo termine di un server Web.

Per correggere e rendere valida anche per il futuro la configurazione TLS

1. Aprire il file di configurazione `/etc/httpd/conf.d/ssl.conf` in un editor di testo e commentare la seguente riga inserendo il carattere `"#"` all'inizio:

```
#SSLProtocol all -SSLv3
```

2. Aggiungere la seguente direttiva:

```
#SSLProtocol all -SSLv3
SSLProtocol -SSLv2 -SSLv3 -TLSv1 -TLSv1.1 +TLSv1.2
```

Questa direttiva disabilita in modo esplicito SSL versioni 2 e 3, nonché TLS versioni 1.0 e 1.1. Il server ora non accetta più connessioni crittografate con client che utilizzano crittografie diverse da TLS 1.2. Le descrizioni dettagliate della direttiva illustrano più chiaramente al lettore la tipologia di configurazione impostata per il server.

Note

La disabilitazione di TLS versioni 1.0 e 1.1 consente di bloccare l'accesso al sito da parte di una piccola percentuale di browser Web non aggiornati.

Per modificare l'elenco delle crittografie consentite

1. Nel file di configurazione `/etc/httpd/conf.d/ssl.conf`, individuare la sezione con la direttiva **SSLCipherSuite** e commentare la riga esistente inserendo il carattere `"#"` all'inizio.

```
#SSLCipherSuite HIGH:MEDIUM:!aNULL:!MD5
```

2. Specificare suite di crittografia esplicite e un ordine di crittografia che dia priorità alla funzione Forward Secrecy e che eviti crittografie non sicure. La direttiva `SSLCipherSuite` qui utilizzata si basa su un output del [generatore di configurazioni SSL di Mozilla](#), che personalizza una configurazione TLS in funzione del software specifico in esecuzione sul server. Per prima cosa determinare le versioni di Apache e OpenSSL in base all'output dei seguenti comandi.

```
[ec2-user ~]$ yum list installed | grep httpd
```

```
[ec2-user ~]$ yum list installed | grep openssl
```

Ad esempio, se l'informazione restituita è Apache 2.4.34 e OpenSSL 1.0.2, inserirla nel generatore. Scegliere poi il modello di compatibilità “moderna”, che crea una direttiva `SSLCipherSuite` e applica in modo rigido la sicurezza ma che funziona per la maggior parte dei browser. Se il software non supporta la configurazione moderna, è possibile aggiornarlo o scegliere la configurazione “intermedia”.

```
SSLCipherSuite ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-
ECDSA-CHACHA20-POLY1305:
ECDHE-RSA-CHACHA20-POLY1305:ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-GCM-
SHA256:
ECDHE-ECDSA-AES256-SHA384:ECDHE-RSA-AES256-SHA384:ECDHE-ECDSA-AES128-SHA256:ECDHE-
RSA-AES128-SHA256
```

Le crittografie selezionate includono nel proprio nome l'acronimo ECDHE (, abbreviazione di Elliptic Curve Diffie-Hellman Ephemeral). Il termine effimero fa riferimento alla proprietà Forward Secrecy. Come sottoprodotto, questi codici non supportano RC4

È consigliabile utilizzare un elenco esplicito di crittografie anziché utilizzare le impostazioni predefinite o le direttive concise il cui contenuto non è visibile.

Copiare la direttiva generata in `/etc/httpd/conf.d/ssl.conf`.

Note

Nonostante in questa sede siano riportate su più righe per facilitarne la leggibilità, una volta copiata su `/etc/httpd/conf.d/ssl.conf` la direttiva deve trovarsi su un'unica riga con solo due punti (senza spazi) tra i nomi di crittografia.

3. Rimuovere infine i commenti mediante la rimozione del carattere “#” dall'inizio della riga:

```
#SSLHonorCipherOrder on
```

Questa direttiva obbliga il server a preferire crittografie con classificazione più elevata, comprese (in questo caso) quelle che supportano la proprietà Forward Secrecy. Con questa direttiva

abilitata, il server cerca di stabilire una connessione stabile e affidabile prima di ripiegare sulle crittografie consentite con un livello inferiore di sicurezza.

Dopo aver completato entrambe le procedure, salvare le modifiche a `/etc/httpd/conf.d/ssl.conf` e riavviare Apache.

Se testate nuovamente il dominio su [Qualys SSL Labs](#), dovrete vedere che la RC4 vulnerabilità e gli altri avvisi sono scomparsi e il riepilogo sarà simile al seguente.

Valutazione complessiva	A
Certificato	100%
Supporto dei protocolli	100%
Scambio di chiavi	90%
Affidabilità crittografia	90%

Ogni aggiornamento a OpenSSL introduce nuove crittografie e rimuove il supporto per quelle vecchie. Conserva la tua AL2023 istanza EC2 up-to-date, tieni d'occhio gli annunci di sicurezza di [OpenSSL](#) e fai attenzione alle segnalazioni di nuovi exploit di sicurezza pubblicate dalla stampa tecnica.

Risoluzione dei problemi

- Il server Web Apache non si avvia a meno che non venga fornita una password

Si tratta del comportamento previsto se per il server hai installato una chiave privata crittografata e protetta con password.

Puoi rimuovere i requisiti di crittografia e password dalla chiave. Supponiamo, ad esempio, di avere una chiave RSA crittografata privata denominata `custom.key` nella directory di default e associata alla password **abcde12345**. Per generare una versione non crittografata della chiave, devi eseguire i seguenti comandi nell'istanza EC2:

```
[ec2-user ~]$ cd /etc/pki/tls/private/
[ec2-user private]$ sudo cp custom.key custom.key.bak
[ec2-user private]$ sudo openssl rsa -in custom.key -passin pass:abcde12345 -out
custom.key.nocrypt
```

```
[ec2-user private]$ sudo mv custom.key.nocrypt custom.key
[ec2-user private]$ sudo chown root:root custom.key
[ec2-user private]$ sudo chmod 600 custom.key
[ec2-user private]$ sudo systemctl restart httpd
```

A questo punto, Apache viene avviato senza visualizzare alcuna richiesta di password.

- Vengono visualizzati errori quando si esegue il comando `sudo yum install -y mod_ssl`.

Quando installi i pacchetti richiesti per SSL, è possibile che vengano visualizzati errori simili ai seguenti.

```
Error: httpd24-tools conflicts with httpd-tools-2.2.34-1.16.amzn1.x86_64
Error: httpd24 conflicts with httpd-2.2.34-1.16.amzn1.x86_64
```

Ciò significa in genere che l'istanza EC2 non è in esecuzione. AL2023 Questo tutorial supporta solo istanze appena create a partire da un'AMI di AL2023 ufficiale.

Tutorial: Ospita un WordPress blog su AL2023

Le seguenti procedure ti aiuteranno a installare, configurare e proteggere un WordPress blog sulla tua istanza AL2023. Questo tutorial è una buona introduzione all'uso di Amazon EC2 in quanto hai il pieno controllo su un server Web che ospita il tuo WordPress blog, cosa non tipica di un servizio di hosting tradizionale.

È tua responsabilità aggiornare i pacchetti software e gestire le patch di sicurezza del server. Per un'WordPress installazione più automatizzata che non richieda l'interazione diretta con la configurazione del server Web, il CloudFormation servizio fornisce un WordPress modello che può anche aiutarti a iniziare rapidamente. Per ulteriori informazioni, consulta [Nozioni di base](#) nella Guida per l'utente di AWS CloudFormation . Se hai bisogno di una soluzione ad alta disponibilità con un database disaccoppiato, consulta [Implementazione di un WordPress sito Web ad alta disponibilità](#) nella Guida per gli sviluppatori.AWS Elastic Beanstalk

Important

Queste procedure sono destinate all'uso con AL2023 Per informazioni su altre distribuzioni, consulta la documentazione specifica. Numerose fasi in questo tutorial non funzionano sulle istanze Ubuntu. Per informazioni WordPress sull'installazione su un'istanza di Ubuntu,

[WordPress](#) consulta la documentazione di Ubuntu. Puoi anche [CodeDeploy](#) utilizzarlo per eseguire questa operazione su sistemi Amazon Linux, macOS o Unix.

Argomenti

- [Prerequisiti](#)
- [Installa WordPress](#)
- [Fasi successive](#)
- [Aiuto! Il nome DNS pubblico è cambiato e il blog non è accessibile](#)

Prerequisiti

Ti consigliamo vivamente di associare un indirizzo IP elastico (EIP) all'istanza che stai utilizzando per ospitare un blog. WordPress Ciò impedisce all'indirizzo DNS pubblico dell'istanza di modificare e interrompere l'installazione. Se sei proprietario di un nome di dominio e vuoi utilizzarlo per il tuo blog, puoi aggiornare il record DNS del nome di dominio in modo che punti all'indirizzo EIP (per ulteriori informazioni su questa procedura, contatta il registrar di nomi di dominio). Puoi usufruire di un indirizzo EIP associato a un'istanza in esecuzione gratuitamente. Per ulteriori informazioni, consulta [Indirizzi IP elastici](#) nella Guida per l'utente di Amazon EC2. Il tutorial [Tutorial: Installare un server LAMP su AL2023](#) include inoltre la procedura per configurare un gruppo di sicurezza che permetta il traffico HTTP e HTTPS, nonché varie fasi da eseguire per verificare che le autorizzazioni di file siano state configurate correttamente per il server Web. Per informazioni sull'aggiunta di regole al gruppo di sicurezza, consulta [Aggiungere regole a un gruppo di sicurezza](#).

Se non disponi ancora di un nome di dominio per il tuo blog, puoi registrare un nome di dominio con Route 53 e associare l'indirizzo EIP dell'istanza al nome di dominio. Per ulteriori informazioni, consulta la pagina relativa alla [registrazione dei nomi di dominio utilizzando Amazon Route 53](#) nella Guida per lo sviluppatore di Amazon Route 53.

Installa WordPress

Connect all'istanza e scarica il pacchetto WordPress di installazione. Per ulteriori informazioni sulla connessione all'istanza, consulta [AL2023 Connessione alle istanze](#).

1. Scarica e installa questi pacchetti usando il comando seguente.

```
dnf install wget php-mysqlnd httpd php-fpm php-mysql cli mariadb105-server php-json
php php-devel -y
```

2. Puoi notare un avviso visualizzato con un messaggio simile nell'output (le versioni possono variare nel tempo):

```
WARNING:
  A newer release of "Amazon Linux" is available.

  Available Versions:

dnf upgrade --releasever=2023.0.20230202

  Release notes:
  https://aws.amazon.com

Version 2023.0.20230204:
  Run the following command to update to 2023.0.20230204:

  dnf upgrade --releasever=2023.0.20230204 ... etc
```

Come best practice, consigliamo di mantenere il sistema operativo il più up-to-date possibile, ma potresti voler ripetere ogni versione per assicurarti che non vi siano conflitti nel tuo ambiente. Se l'installazione dei pacchetti precedenti indicati al passaggio 1 ha esito negativo, potrebbe essere necessario eseguire l'aggiornamento a una delle versioni più recenti elencate e riprovare.

3. Scaricate il pacchetto di WordPress installazione più recente con il wget comando. Il comando seguente dovrebbe scaricare sempre la versione più recente.

```
[ec2-user ~]$ wget https://wordpress.org/latest.tar.gz
```

4. Decomprimere ed estrarre il pacchetto di installazione. La cartella di installazione viene decompressa in una cartella denominata wordpress.

```
[ec2-user ~]$ tar -xzf latest.tar.gz
```

Per creare un utente del database e un database per l' WordPress installazione

WordPress L'installazione deve archiviare informazioni, come post di blog e commenti degli utenti, in un database. Questa procedura consente di creare un database del blog e un utente autorizzato a leggere e salvare le informazioni in tale database.

1. Avvia il server di database e il server Web.

```
[ec2-user ~]$ sudo systemctl start mariadb httpd
```

2. Accedere al server di database come utente root. Immetti la password database root quando viene richiesto. Questa password potrebbe essere diversa dalla password del sistema root oppure potrebbe anche essere vuota se non hai impostato alcuna protezione per il server di database.

Se non hai ancora definito la protezione del server di database, è importante che tu lo faccia ora. Per ulteriori informazioni, vedere [Fase 3: proteggere il server di database](#) (AL2023).

```
[ec2-user ~]$ mysql -u root -p
```

3. Creare un utente e una password per il database MySQL. L' WordPress installazione utilizza questi valori per comunicare con il database MySQL. Immettere il seguente comando, ricordandosi di sostituire gli argomenti con un nome utente univoco e una password.

```
CREATE USER 'wordpress-user'@'localhost' IDENTIFIED BY 'your_strong_password';
```

Assicurarsi di creare una password complessa per l'utente. Non utilizzare l'apostrofo (') nella password perché interromperebbe l'esecuzione del comando che lo precede. Non riutilizzare una password esistente e accertarsi di memorizzare questa password in un luogo sicuro.

4. Creare il database. Assegnare al database un nome descrittivo e significativo, ad esempio `wordpress-db`.

Note

I segni di punteggiatura che racchiudono il nome del database nel comando riportato di seguito sono definiti backtick (apice rovescio). Il tasto del segno backtick (apice rovescio) (`) in genere si trova sopra il tasto Tab su una tastiera standard. I backtick

non sono sempre richiesti, ma consentono di utilizzare caratteri altrimenti non validi, ad esempio i trattini, nei nomi di database.

```
CREATE DATABASE `wordpress-db`;
```

5. Concedi i privilegi completi per il tuo database all' WordPress utente che hai creato in precedenza.

```
GRANT ALL PRIVILEGES ON `wordpress-db`.* TO "wordpress-user"@"localhost";
```

6. Scaricare i privilegi del database per implementare tutte le modifiche apportate.

```
FLUSH PRIVILEGES;
```

7. Uscire dal client mysql.

```
exit
```

Per creare e modificare il file wp-config.php

La cartella WordPress di installazione contiene un file di configurazione di esempio chiamato wp-config-sample.php. In questa procedura, puoi copiare questo file e modificarlo in modo conforme a una configurazione specifica.

1. Copiare il file wp-config-sample.php in un file denominato wp-config.php. In questo modo, crei un nuovo file di configurazione mantenendo intatto il file campione originale come backup.

```
[ec2-user ~]$ cp wordpress/wp-config-sample.php wordpress/wp-config.php
```

2. Modificare il file wp-config.php con l'editor di testo preferito (ad esempio nano o vim) e immettere i valori dell'installazione in uso. Se non si dispone di un editor di testo preferito, nano è adatto agli utenti non esperti.

```
[ec2-user ~]$ nano wordpress/wp-config.php
```

- a. Cercare la riga che definisce `DB_NAME` e modificare `database_name_here` utilizzando il nome di database creato in [Step 4](#) di [Per creare un utente del database e un database per l'WordPress installazione](#).

```
define('DB_NAME', 'wordpress-db');
```

- b. Cercare la riga che definisce `DB_USER` e modificare `username_here` utilizzando l'utente database creato in [Step 3](#) di [Per creare un utente del database e un database per l'WordPress installazione](#).

```
define('DB_USER', 'wordpress-user');
```

- c. Cercare la riga che definisce `DB_PASSWORD` e modificare `password_here` utilizzando la password complessa creata in [Step 3](#) di [Per creare un utente del database e un database per l'WordPress installazione](#).

```
define('DB_PASSWORD', 'your_strong_password');
```

- d. Cercare la sezione denominata Authentication Unique Keys and Salts. Questi KEY e questi SALT valori forniscono un livello di crittografia ai cookie del browser che WordPress gli utenti archiviano sui loro computer locali. In sostanza, l'aggiunta di valori lunghi e casuali rende il sito più sicuro. Visita <https://api.wordpress.org/secret-key/1.1/salt/> per generare in modo casuale un set di valori chiave che puoi copiare e incollare nel tuo `wp-config.php` file. Per incollare il testo in un'applicazione terminale PuTTY, posizionare il cursore nel punto in cui si desidera incollare il testo e fare clic con il pulsante destro del mouse all'interno dell'applicazione terminale PuTTY.

[Per ulteriori informazioni sulle chiavi di sicurezza, visita https://wordpress.org/support/article/editing-wp-config-php/#security-keys](https://wordpress.org/support/article/editing-wp-config-php/#security-keys).

Note

I valori riportati di seguito sono a solo scopo di esempio. Non utilizzarli per l'installazione in uso.

```
define('AUTH_KEY', ' #U$$+[RXN8:b^-L 0(WU_+ c+WFkI~c]o]-bHw+)/Aj[wTwSiZ<Qb[mghEXcRh-');
```

```
define('SECURE_AUTH_KEY', 'Zsz._P=l/|y.Lq)XjlkwS1y5NJ76E6EJ.AV0pCKZZB,*~*r ?
60P$eJT@;+(ndLg');
define('LOGGED_IN_KEY', 'ju}qwre3V*+8f_z0Wf?{LlGsQ]Ye@2Jh^,8x>)Y |;(^[Iw]Pi
+LG#A4R?7N`YB3');
define('NONCE_KEY', 'P(g62HeZxEes|LnI^i=H,[XwK9I&[2s|:~?0N}VJM%?;v2v]v+;
+^9eXUahg@:~Cj');
define('AUTH_SALT', 'C$DpB4Hj[JK:~{qL`sRvA:~{~7yShy(9A@5wg+`JJVb1fk%_-
Bx*M4(qc[Qg%JT!h');
define('SECURE_AUTH_SALT', 'd!uRu#}+q#{f$Z?Z9uFPG.$~{+S{n~1M&%@~gL>U>NV<zpD-@2-
Es7Q10-bp28EKv');
define('LOGGED_IN_SALT', ';j{00P*owZf)kVD+FVLn-~ >.|Y%Ug4#I^*LVd9QeZ^&XmK|
e(76miC+&W&+^0P/');
define('NONCE_SALT', '-97r*V/cgxLmp?Zy4zUU4r99QQ_rGs2LTd%P;|
_e1tS)8_B/,~.6[=UK<J_y9?JWG');
```

- e. Salva il file ed esci dall'editor di testo.

Per installare i WordPress file nella cartella principale del documento Apache

- Dopo aver decompresso la cartella di installazione, creato un database e un utente MySQL e personalizzato il file di WordPress configurazione, è possibile copiare i file di installazione nella cartella principale dei documenti del server Web in modo da poter eseguire lo script di installazione che completa l'installazione. La posizione di questi file dipende dal fatto che il WordPress blog sia disponibile nella directory principale effettiva del server Web (ad esempio, *my.public.dns.amazonaws.com*) o in una sottodirectory o cartella sotto la radice (ad esempio, *my.public.dns.amazonaws.com/blog*).
- Se volete WordPress eseguirlo nella directory principale del documento, copiate il contenuto della directory di installazione di wordpress (ma non la directory stessa) come segue:

```
[ec2-user ~]$ cp -r wordpress/* /var/www/html/
```

- Se volete WordPress eseguirlo in una directory alternativa sotto la radice del documento, create prima quella directory e poi copiate i file al suo interno. In questo esempio, WordPress verrà eseguito dalla directory `blog`:

```
[ec2-user ~]$ mkdir /var/www/html/blog
[ec2-user ~]$ cp -r wordpress/* /var/www/html/blog/
```

⚠ Important

Per motivi di sicurezza, se non si passa immediatamente alla procedura successiva, arrestare ora il server Web Apache (`httpd`). Dopo aver spostato l'installazione nella directory principale del documento Apache, lo script di WordPress installazione non è protetto e un utente malintenzionato potrebbe accedere al tuo blog se il server web Apache fosse in esecuzione. Per arrestare il server Web Apache, immettere il comando `sudo service httpd stop`. Se invece si passa alla procedura successiva, non è necessario arrestare il server Web Apache.

Per consentire l'uso dei permalink WordPress

WordPress i permalink devono utilizzare i `.htaccess` file Apache per funzionare correttamente, ma questo non è abilitato di default su Amazon Linux. Utilizza la seguente procedura per consentire tutte le modifiche nella directory radice dei documenti di Apache.

1. Aprire il file `httpd.conf` con l'editor di testo preferito (ad esempio `nano` o `vim`). Se non si dispone di un editor di testo preferito, `nano` è adatto agli utenti non esperti.

```
[ec2-user ~]$ sudo vim /etc/httpd/conf/httpd.conf
```

2. Cercare la sezione che inizia con `<Directory "/var/www/html">`.

```
<Directory "/var/www/html">
#
# Possible values for the Options directive are "None", "All",
# or any combination of:
#   Indexes Includes FollowSymLinks SymLinksifOwnerMatch ExecCGI MultiViews
#
# Note that "MultiViews" must be named *explicitly* --- "Options All"
# doesn't give it to you.
#
# The Options directive is both complicated and important. Please see
# http://httpd.apache.org/docs/2.4/mod/core.html#options
# for more information.
#
Options Indexes FollowSymLinks

#
# AllowOverride controls what directives may be placed in .htaccess files.
```

```
# It can be "All", "None", or any combination of the keywords:
# Options FileInfo AuthConfig Limit
#
AllowOverride None

#
# Controls who can get stuff from this server.
#
Require all granted
</Directory>
```

3. Modificare la riga `AllowOverride None` nella sezione precedente in modo che sia impostata nel seguente modo: `AllowOverride ALL`.

Note

Sono presenti più righe `AllowOverride` in questo file. Assicurarsi di modificare la riga nella sezione `<Directory "/var/www/html">`.

```
AllowOverride ALL
```

4. Salva il file ed esci dall'editor di testo.

Per installare la libreria di disegni grafici PHP su AL2023

La libreria GD per PHP consente di modificare le immagini. Installa questa libreria se hai bisogno di ritagliare l'immagine di intestazione per il tuo blog. La versione da installare potrebbe richiedere una versione minima specifica di questa libreria (ad esempio, la versione 8.1). phpMyAdmin

Usa il seguente comando per installare la libreria di disegni grafici PHP su AL2023. Ad esempio, se hai installato php8.1 dall'origine come parte dell'installazione dello stack LAMP, questo comando installa la versione 8.1 della libreria di disegni grafici PHP.

```
[ec2-user ~]$ sudo dnf install php-gd
```

Per verificare la versione installata utilizza il seguente comando:

```
[ec2-user ~]$ sudo dnf list installed | grep php-gd
```

Di seguito è riportato un output di esempio:

```
php-gd.x86_64                8.1.30-1.amzn2                @amazonlinux
```

Per installare la libreria di disegni grafici PHP nell'Amazon Linux AMI

La libreria GD per PHP consente di modificare le immagini. Installa questa libreria se hai bisogno di ritagliare l'immagine di intestazione per il tuo blog. La versione phpMyAdmin che installate potrebbe richiedere una versione minima specifica di questa libreria (ad esempio, la versione 8.1).

Per verificare quali versioni sono disponibili utilizza il comando seguente:

```
[ec2-user ~]$ dnf list | grep php
```

Di seguito sono riportate righe di esempio dell'output per la libreria di disegni grafici PHP (versione 8.1):

```
php8.1.aarch64                8.1.7-1.amzn2023.0.1
php8.1.aarch64                @amazonlinux
php8.1-cli.aarch64           8.1.7-1.amzn2023.0.1
php8.1-cli.aarch64           @amazonlinux
php8.1-common.aarch64       8.1.7-1.amzn2023.0.1
php8.1-common.aarch64       @amazonlinux
php8.1-devel.aarch64        8.1.7-1.amzn2023.0.1
php8.1-devel.aarch64        @amazonlinux
php8.1-fpm.aarch64          8.1.7-1.amzn2023.0.1
php8.1-fpm.aarch64          @amazonlinux
php8.1-gd.aarch64           8.1.7-1.amzn2023.0.1
php8.1-gd.aarch64           @amazonlinux
```

Utilizza il seguente comando per installare una versione specifica della libreria di disegni grafici PHP (ad esempio, versione php8.1) nell'AMI Amazon Linux:

```
[ec2-user ~]$ sudo dnf install -y php8.1-gd
```

Per correggere le autorizzazioni di file sul server Web Apache

Alcune delle funzionalità disponibili in Apache WordPress richiedono l'accesso in scrittura alla radice del documento Apache (come il caricamento di contenuti multimediali tramite le schermate di amministrazione). Se non già stato fatto, applicare le seguenti appartenenze e autorizzazioni di

gruppo, come descritto con maggiore dettaglio in [Tutorial: installa un server Web LAMP con Amazon Linux AMI](#).

1. Garantire la proprietà dei file di `/var/www` e dei suoi contenuti all'utente `apache`.

```
[ec2-user ~]$ sudo chown -R apache /var/www
```

2. Garantire la proprietà del gruppo di `/var/www` e dei suoi contenuti al gruppo `apache`.

```
[ec2-user ~]$ sudo chgrp -R apache /var/www
```

3. Modificare le autorizzazioni a livello di directory di `/var/www` e delle relative sottodirectory per aggiungere le autorizzazioni di scrittura e impostare l'ID gruppo per le sottodirectory future.

```
[ec2-user ~]$ sudo chmod 2775 /var/www
[ec2-user ~]$ find /var/www -type d -exec sudo chmod 2775 {} \;
```

4. Modifica in modo ricorsivo le autorizzazioni di file di `/var/www` e delle relative sottodirectory.

```
[ec2-user ~]$ find /var/www -type f -exec sudo chmod 0644 {} \;
```

Note

Se intendi utilizzarlo anche WordPress come server FTP, qui avrai bisogno di impostazioni di gruppo più permissive. Per eseguire questa operazione, consulta [i passaggi e le impostazioni di sicurezza consigliati WordPress in](#).

5. Riavviare il server Web Apache per implementare il nuovo gruppo e le nuove autorizzazioni.

```
[ec2-user ~]$ sudo systemctl restart httpd
```

Per eseguire lo script WordPress di installazione con AL2023

Sei pronto per l'installazione WordPress. I comandi utilizzati dipendono dal sistema operativo. I comandi di questa procedura possono essere utilizzati con AL2023. Usa la procedura che segue questa con AL2023 AMI.

1. Utilizzare il comando `systemctl` per assicurarsi che i servizio `httpd` e di database vengano avviati a ogni avvio del sistema.

```
[ec2-user ~]$ sudo systemctl enable httpd && sudo systemctl enable mariadb
```

2. Verificare che il server di database sia in esecuzione.

```
[ec2-user ~]$ sudo systemctl status mariadb
```

Se il servizio di database non è in esecuzione, avviarlo.

```
[ec2-user ~]$ sudo systemctl start mariadb
```

3. Verificare che il server Web Apache (httpd) sia in esecuzione.

```
[ec2-user ~]$ sudo systemctl status httpd
```

Se il servizio httpd non è in esecuzione, avviarlo.

```
[ec2-user ~]$ sudo systemctl start httpd
```

4. In un browser web, digita l'URL del tuo WordPress blog (l'indirizzo DNS pubblico dell'istanza o l'indirizzo seguito dalla blog cartella). Dovresti vedere lo script di WordPress installazione. Fornisci le informazioni richieste dall' WordPress installazione. Per completare l'installazione, seleziona Installa WordPress. Per ulteriori informazioni, consulta [Passaggio 5: Esecuzione dello script di installazione](#) sul WordPress sito Web.

Per eseguire lo script WordPress di installazione con AL2023 AMI

1. Utilizzare il comando chkconfig per assicurarsi che i servizio httpd e di database vengano avviati a ogni avvio del sistema.

```
[ec2-user ~]$ sudo chkconfig httpd on && sudo chkconfig mariadb on
```

2. Verificare che il server di database sia in esecuzione.

```
[ec2-user ~]$ sudo service mariadb status
```

Se il servizio di database non è in esecuzione, avviarlo.

```
[ec2-user ~]$ sudo service mariadb start
```

3. Verificare che il server Web Apache (httpd) sia in esecuzione.

```
[ec2-user ~]$ sudo service httpd status
```

Se il servizio httpd non è in esecuzione, avviarlo.

```
[ec2-user ~]$ sudo service httpd start
```

4. In un browser web, digita l'URL del tuo WordPress blog (l'indirizzo DNS pubblico dell'istanza o l'indirizzo seguito dalla `blog` cartella). Dovresti vedere lo script di WordPress installazione. Fornisci le informazioni richieste dall' WordPress installazione. Per completare l'installazione, seleziona Installa WordPress. Per ulteriori informazioni, consulta [Passaggio 5: Esecuzione dello script di installazione](#) sul WordPress sito Web.

Fasi successive

Dopo aver testato il tuo WordPress blog, valuta la possibilità di aggiornarne la configurazione.

Utilizza un nome di dominio personalizzato

Se all'indirizzo EIP dell'istanza EC2 è associato un nome di dominio, puoi configurare il blog in modo che utilizzi tale nome anziché l'indirizzo DNS EC2 pubblico. Per ulteriori informazioni, consulta [Modifica dell'URL del sito](#) sul WordPress sito Web.

Configurazione del blog

Puoi configurare il blog in modo che utilizzi [temi](#) e [plugin](#) diversi in modo da offrire un'esperienza più personalizzata ai lettori. Tuttavia, il processo di installazione può talvolta generare problemi che portano alla perdita dell'intero blog. Pertanto, consigliamo vivamente di eseguire una copia di backup dell'Amazon Machine Image (AMI) dell'istanza prima di tentare di installare temi o plugin in modo da essere in grado di ripristinare il blog in caso di problemi durante l'installazione. Per ulteriori informazioni, consulta [Crea la tua AMI](#) nella Guida per l'utente di Amazon EC2.

Aumento della capacità

Se il tuo WordPress blog diventa popolare e hai bisogno di maggiore potenza di calcolo o spazio di archiviazione, considera i seguenti passaggi:

- Espandi lo spazio di storage sull'istanza. Per ulteriori informazioni, consulta [Elastic Volumes Amazon EBS](#).
- Trasferisci il database MySQL in [Amazon RDS](#) in modo da sfruttare tutte le funzionalità di scalabilità del servizio.

Miglioramento delle prestazioni di rete del traffico Internet

Se ti aspetti che il tuo blog gestisca il traffico da parte di utenti situati in tutto il mondo, considera l'uso di [AWS Global Accelerator](#). Global Accelerator ti aiuta a ridurre la latenza migliorando le prestazioni del traffico Internet tra i dispositivi client degli utenti e l'applicazione su cui è in esecuzione WordPress. AWS Global Accelerator utilizza la [rete AWS globale](#) per indirizzare il traffico verso un endpoint applicativo funzionante nella AWS regione più vicina al client.

Scopri di più su WordPress

I seguenti collegamenti contengono ulteriori informazioni su WordPress.

- Per informazioni in merito WordPress, consultate la documentazione di aiuto del WordPress Codex disponibile su [Codex](#).
- Per ulteriori informazioni sulla risoluzione dei problemi di installazione, consulta [Problemi di installazione comuni](#).
- [Per informazioni su come rendere il tuo WordPress blog più sicuro, consulta Hardening. WordPress](#)
- Per informazioni sulla gestione del WordPress blog up-to-date, consulta [Aggiornamento WordPress](#).

Aiuto! Il nome DNS pubblico è cambiato e il blog non è accessibile

L' WordPress installazione viene configurata automaticamente utilizzando l'indirizzo DNS pubblico per l'istanza EC2. Se arresti e riavvii l'istanza, l'indirizzo DNS pubblico cambia, a meno che non sia associato a un indirizzo IP elastico, e il blog non funzionerà più perché fa riferimento a risorse disponibili in un indirizzo che non esiste più o che è assegnato a un'altra istanza EC2. [Una descrizione più dettagliata del problema e diverse possibili soluzioni sono disponibili in https://wordpress.org/support/article/changing-the-site-url/](https://wordpress.org/support/article/changing-the-site-url/).

Se ciò si è verificato durante l' WordPress installazione, potrebbe essere possibile ripristinare il blog seguendo la procedura riportata di seguito, che utilizza l'interfaccia a riga di wp-cli comando per WordPress.

Per modificare l'URL del WordPress sito con wp-cli

1. Connettersi all'istanza EC2 con SSH.
2. Annotare il vecchio URL del sito e il nuovo URL del sito relativi all'istanza. Il vecchio URL del sito è probabilmente il nome DNS pubblico dell'istanza EC2 al momento dell'installazione. WordPress È possibile che il nuovo URL del sito sia il nome DNS pubblico corrente per l'istanza EC2. Se non sei certo del vecchio URL del sito, puoi utilizzare curl per cercarlo utilizzando il seguente comando.

```
[ec2-user ~]$ curl localhost | grep wp-content
```

I riferimenti al vecchio nome DNS pubblico dovrebbero essere presenti nell'output e sono simili a quanto segue (il vecchio URL del sito è visualizzato in rosso):

```
<script type='text/javascript' src='http://ec2-52-8-139-223.us-west-1.compute.amazonaws.com/wp-content/themes/twentyfifteen/js/functions.js?ver=20150330'></script>
```

3. Scaricare wp-cli con il seguente comando.

```
[ec2-user ~]$ curl -O https://raw.githubusercontent.com/wp-cli/builds/gh-pages/phar/wp-cli.phar
```

4. Cerca e sostituisci il vecchio URL del sito nell' WordPress installazione con il seguente comando. Sostituisci il vecchio e il nuovo URL del sito con l'istanza EC2 e il percorso dell' WordPress installazione (di solito /var/www/html o) /var/www/html/blog

```
[ec2-user ~]$ php wp-cli.phar search-replace 'old_site_url' 'new_site_url' --path=/path/to/wordpress/installation --skip-columns=guid
```

5. In un browser web, inserisci il nuovo URL del sito del tuo WordPress blog per verificare che il sito funzioni di nuovo correttamente. In caso contrario, consulta [Modifica dell'URL del sito](#) e [Problemi di installazione comuni](#) per ulteriori informazioni.

Tutorial: transizione da Redis 6 a Valkey su 023 AL2

La seguente documentazione descrive gli aspetti chiave della transizione da Redis 6 a Valkey su 023 AL2

Cronologia del supporto per Redis 6

Redis 6 raggiungerà la fine del ciclo di vita (EOL) il 31 gennaio 2027. Dopo questa data, Redis 6 non riceverà più aggiornamenti o patch di sicurezza dal progetto Redis. Consigliamo vivamente agli utenti di migrare a Valkey prima di gennaio 2027 per garantire supporto e aggiornamenti di sicurezza continui.

[Per ulteriori informazioni sulle tempistiche di supporto delle versioni Redis, consulta la documentazione di Redis Schedule. End-Of-Life](#)

Introduzione a Valkey

Valkey è un fork open source di Redis 7, gestito da The Linux Foundation. È completamente compatibile con le versioni del software Redis Open Source (OSS) dalla 2.x alla 7.2.x. Valkey mantiene l'API e le funzionalità familiari di Redis, offrendo al contempo diversi miglioramenti:

- Prestazioni migliorate grazie al multithreading.
- Migliore efficienza della memoria, specialmente in modalità cluster.
- Replica a doppio canale per una migliore coerenza dei dati.

Piano e tempistica della migrazione

Gli utenti sono fortemente incoraggiati a migrare da Redis 6 a Valkey prima del 31 gennaio 2027, quando Redis 6 raggiungerà la fine del ciclo di vita (EOL). Questa migrazione richiede un intervento manuale e non è automatica.

Amazon Linux consiglia questa migrazione per garantire funzionalità, supporto e aggiornamenti di sicurezza continui per le applicazioni dipendenti da Redis.

Opzioni e passaggi di migrazione

Proponiamo tre percorsi di migrazione verso Valkey in base ai requisiti di implementazione e alle esigenze operative.

Opzione 1: installazione di una nuova istanza

Per nuove implementazioni o quando non è necessaria la migrazione dei dati:

1. Installa Valkey:

```
[ec2-user ~]$ sudo dnf install valkey
```

2. Avvia Valkey:

```
[ec2-user ~]$ sudo systemctl start valkey
```

3. (Facoltativo) Abilita Valkey all'avvio:

```
[ec2-user ~]$ sudo systemctl enable valkey
```

4. Verifica l'installazione:

```
[ec2-user ~]$ valkey-cli info server  
[ec2-user ~]$ valkey-cli ping
```

Opzione 2: sostituzione sul posto

Per i casi esistenti in cui non è richiesta la persistenza dei dati:

1. Arresta Redis 6:

```
[ec2-user ~]$ sudo systemctl stop redis6
```

2. Installa Valkey:

```
[ec2-user ~]$ sudo dnf install valkey
```

3. (Opzionale) Usa la configurazione Redis 6 in Valkey:

```
[ec2-user ~]$ sudo cp /etc/redis6/redis6.conf /etc/valkey/valkey.conf  
[ec2-user ~]$ sudo cp /etc/valkey/valkey.conf /etc/valkey/valkey.conf.backup  
[ec2-user ~]$ sudo chown valkey:root /etc/valkey/valkey.conf  
[ec2-user ~]$ sudo sed -i 's|^dir\s.*|dir /var/lib/valkey|g' /etc/valkey/  
valkey.conf
```

4. (Facoltativo) Usa il file di configurazione Redis 6 sentinel in Valkey:

```
[ec2-user ~]$ sudo cp /etc/redis6/sentinel.conf /etc/valkey/sentinel.conf
```

```
[ec2-user ~]$ sudo chown valkey:root /etc/valkey/sentinel.conf
```

5. Avvia Valkey:

```
[ec2-user ~]$ sudo systemctl start valkey
```

6. (Facoltativo) Abilita Valkey all'avvio:

```
[ec2-user ~]$ sudo systemctl enable valkey
```

7. Verifica l'installazione di Valkey:

```
[ec2-user ~]$ valkey-cli info server  
[ec2-user ~]$ valkey-cli ping
```

8. Rimuovi Redis 6:

```
[ec2-user ~]$ sudo dnf remove redis6
```

Opzione 3: migrazione dei dati

Questa opzione consente di eseguire contemporaneamente Redis 6 e Valkey.

1. Installa Valkey senza rimuovere Redis 6:

```
[ec2-user ~]$ sudo dnf install valkey
```

2. (Facoltativo) Usa la configurazione Redis 6 in Valkey:

```
[ec2-user ~]$ sudo cp /etc/redis6/redis6.conf /etc/valkey/valkey.conf  
[ec2-user ~]$ sudo cp /etc/valkey/valkey.conf /etc/valkey/valkey.conf.backup  
[ec2-user ~]$ sudo chown valkey:root /etc/valkey/valkey.conf  
[ec2-user ~]$ sudo sed -i 's|^dir\s.*|dir /var/lib/valkey|g' /etc/valkey/  
valkey.conf
```

3. (Facoltativo) Usa il file di configurazione Redis 6 sentinel in Valkey:

```
[ec2-user ~]$ sudo cp /etc/redis6/sentinel.conf /etc/valkey/sentinel.conf  
[ec2-user ~]$ sudo chown valkey:root /etc/valkey/sentinel.conf
```

4. Modifica la configurazione di Valkey:

Modifica `/etc/valkey/valkey.conf` e imposta la direttiva `'port'` su un valore diverso (ad esempio, 6380) per evitare conflitti con Redis 6.

5. Avvia Valkey:


```
[ec2-user ~]$ sudo systemctl start valkey
```

6. (Facoltativo) Abilita Valkey all'avvio:

```
[ec2-user ~]$ sudo systemctl enable valkey
```

7. Verifica l'installazione di Valkey:

```
[ec2-user ~]$ valkey-cli -p port info server  
[ec2-user ~]$ valkey-cli -p port ping
```

 Note

Sostituisci *port* con il numero di porta configurato.

8. Migrazione dei dati:

Ora puoi migrare i dati da Redis 6 a Valkey utilizzando metodi di replica o trasferimento manuale dei dati.

9. Aggiorna le configurazioni delle applicazioni:

Aggiorna gradualmente le tue applicazioni per utilizzare la porta Valkey.

10. Rimuovi Redis 6:

Una volta migrati tutti i dati e le applicazioni, puoi interrompere e rimuovere Redis 6.

```
[ec2-user ~]$ sudo systemctl stop redis6  
[ec2-user ~]$ sudo dnf remove redis6
```

Note

Si consiglia vivamente di convalidare il processo di migrazione in un ambiente di test prima di implementare modifiche nei sistemi di produzione.

Argomenti correlati

Per ulteriori informazioni su Valkey:

- [Valkey: https://valkey.io/](https://valkey.io/)
- [Migrazione Valkey: https://valkey.io/topics/migration/](https://valkey.io/topics/migration/)

Tutorial: Installare l'ambiente desktop GNOME su AL2023

L'[ambiente desktop GNOME](#) è disponibile come interfaccia utente grafica opzionale a partire dalla versione AL2023 2023.7 o successiva.

Le seguenti procedure consentono di installare l'ambiente desktop GNOME sulla propria istanza. AL2023 È possibile utilizzare questa interfaccia grafica per interagire con il sistema Linux utilizzando un ambiente desktop familiare anziché solo l'interfaccia a riga di comando.

Indice

- [Prerequisiti](#)
- [Installazione](#)
- [Argomenti correlati](#)

Prerequisiti

- L'ambiente desktop richiede almeno 2,4 GB di memoria. Pertanto, si consiglia un'istanza di tipo t2.medium o superiore per garantire prestazioni adeguate. Esempi di tipi di istanze con memoria insufficiente includono t2.nano, t2.micro, et2.small. Questa restrizione si applica anche alle t3 t4 istanze di queste dimensioni, nonché a qualsiasi altro tipo di istanza che non soddisfa i requisiti di memoria.

- Questo tutorial presuppone che tu abbia già avviato un'istanza utilizzando la versione AL2023 2023.7 o successiva. Per ulteriori informazioni, consulta le pagine and. [AL2023 su Amazon EC2 Aggiornamento AL2023](#)

Installazione

- Installa l'ambiente desktop GNOME e i pacchetti correlati.

```
[ec2-user ~]$ sudo dnf groupinstall "Desktop" -y
```

Note

Per accedere all'ambiente desktop grafico, dovrai installare e configurare software aggiuntivo come Amazon DCV o VNC. Questi strumenti consentono di connettersi e interagire con l'interfaccia utente grafica sulla rete.

Argomenti correlati

Per ulteriori informazioni sull'ambiente desktop grafico, consulta la seguente documentazione:

- [Che cos'è Amazon DCV?](#) nella Amazon DCV Administrator Guide
- [Tutorial: configurare il server TigerVNC su AL2023](#)

Tutorial: configurare il server TigerVNC su AL2023

Le seguenti procedure ti aiutano a configurare il server VNC sulla tua istanza. AL2023 VNC consente di accedere e interagire in remoto con l'ambiente desktop grafico tramite una connessione di rete sicura.

Indice

- [Prerequisiti](#)
- [Fase 1: Installazione](#)
- [Fase 2: Configurazione](#)
- [Passaggio 3: Connect utilizzando un client VNC](#)

- [\(Facoltativo\) Avvia il servizio all'avvio](#)
- [\(Facoltativo\) Disattiva la schermata di blocco inattiva](#)
- [Argomenti correlati](#)

Prerequisiti

- Questo tutorial presuppone che tu abbia già installato l'ambiente desktop GNOME sulla tua istanza. AL2023 Per ulteriori informazioni, consulta la [Tutorial: Installare l'ambiente desktop GNOME su AL2023](#) pagina.
- Questo tutorial utilizza il port forwarding SSH per accedere al server VNC. Per ulteriori informazioni sulla configurazione della tua coppia di key pair, consulta [Connect alla tua istanza Linux usando SSH](#) nella Amazon EC2 User Guide.
- La seguente procedura non ti guida attraverso il processo di installazione di un client VNC. È necessario disporre di un client VNC installato sul computer locale per potersi connettere e interagire con l'ambiente desktop.

Fase 1: Installazione

1. Connettiti alla tua istanza. Per ulteriori informazioni, consulta [AL2023 Connessione alle istanze](#).
2. Installa il pacchetto server TigerVNC per. AL2023

L'-yopzione installa il pacchetto senza chiedere conferma. Se desideri esaminare il pacchetto prima dell'installazione, puoi omettere questa opzione.

```
[ec2-user ~]$ sudo dnf install -y tigervnc-server
```

Fase 2: Configurazione

1. Assicurati che l'utente abbia configurato una password VNC.

```
[ec2-user ~]$ vncpasswd
```

2. Assegna un numero di visualizzazione all'utente.

```
[ec2-user ~]$ sudo vi /etc/tigervnc/vncserver.users
```

Aggiungete la seguente configurazione:

```
:1=ec2-user
```

Note

È possibile assegnare qualsiasi numero di visualizzazione all'utente. Stiamo usando il `display :1` per questo esempio.

3. Modifica il file di configurazione del server VNC.

```
[ec2-user ~]$ sudo vi /etc/tigervnc/vncserver-config-defaults
```

Aggiungi la seguente configurazione:

```
session=gnome
securitytypes=vncauth,tlsvnc
geometry=1920x1080
localhost
alwaysshared
```

Note

È possibile modificare la risoluzione dello schermo utilizzando il `geometry` parametro. Stiamo usando `1920x1080` per questo esempio.

4. Avvia il server VNC. Questo processo deve essere ripetuto ogni volta che riavvii l'istanza. Se desideri automatizzare il processo di avvio di questo servizio, consulta la sezione opzionale riportata di seguito.

```
[ec2-user ~]$ sudo systemctl start vncserver@:1
```

Important

All'avvio del `vncserver` servizio, la parte successiva `@` deve corrispondere al numero visualizzato per l'utente nel `/etc/tigervnc/vncserver.users` file.

Dopo aver eseguito questo passaggio, è possibile creare il tunnel SSH dal computer locale e connettersi utilizzando il client VNC.

Passaggio 3: Connect utilizzando un client VNC

Il server VNC espone un socket TCP per le connessioni client. Sebbene sia possibile esporre la porta VNC direttamente tramite il gruppo di sicurezza, questo tutorial dimostra l'utilizzo del tunneling SSH come approccio più sicuro crittografando la connessione tra il computer locale e l'istanza EC2. Una volta connesso tramite il tunnel, ti autenticherai sul server VNC utilizzando la password che hai configurato nel passaggio precedente. Per ulteriori informazioni sui gruppi di sicurezza, consulta [Modifica dei gruppi di sicurezza per l'istanza Amazon EC2](#) nella Guida per l'utente di Amazon EC2.

1. Crea un tunnel SSH dal tuo computer locale.

```
$ ssh -i <keypair> -L 5901:localhost:5901 ec2-user@<address>
```

Note

Sostituiscilo `<keypair>` con il percorso della tua chiave SSH e `<address>` con l'IP pubblico o il nome DNS dell'istanza. La porta cambia in base al numero di display utilizzato per avviare il `vncserver`. Ad esempio, il display `:1` utilizza la porta `5901`, il display `:2` utilizza la porta `5902`, ecc.

2. Usa il tuo client VNC per connetterti a `localhost:5901` o `127.0.0.1:5901` con la password VNC precedentemente impostata.

Important

Mantieni aperto il tunnel SSH mentre usi VNC. Se il tunnel SSH non è aperto, non sarà possibile utilizzare il client VNC per visualizzare e interagire con l'ambiente desktop.

(Facoltativo) Avvia il servizio all'avvio

Se prevedi di utilizzare VNC regolarmente, potresti voler configurare il server VNC in modo che si avvii automaticamente all'avvio dell'istanza. In questo modo si elimina la necessità di avviare

manualmente il server VNC ogni volta che si riavvia l'istanza. Questa configurazione assicura che l'ambiente desktop grafico sia pronto e accessibile non appena l'istanza completa il processo di avvio.

- Configura il servizio in modo che venga avviato all'avvio.

```
[ec2-user ~]$ sudo systemctl enable vncserver@:1
```

Important

Quando si abilita il `vncserver` servizio, la parte successiva `@` deve corrispondere al numero di visualizzazione impostato per l'utente nel `/etc/tigervnc/vncserver.users` file. Inoltre, è possibile passare l' `--nowargomento` dopo `enable` per avviare immediatamente il servizio.

Dopo aver eseguito questo passaggio, non sarà più necessario avviare l'istanza `vncserver` ogni volta che si riavvia l'istanza.

(Facoltativo) Disattiva la schermata di blocco inattiva

- Imposta il ritardo di inattività su zero per disabilitare la schermata di blocco quando l'utente è inattivo per un periodo di tempo più lungo.

```
[ec2-user ~]$ gsettings set org.gnome.desktop.session idle-delay 0
```

Argomenti correlati

Per ulteriori informazioni sull'ambiente desktop grafico, consulta la seguente documentazione:

- [Tutorial: Installare l'ambiente desktop GNOME su AL2023](#)
- [Che cos'è Amazon DCV?](#) nella Amazon DCV Administrator Guide

Utilizzo di Multi-Gen LRU (MGLRU) sui kernel AL2023

Il [Multi-Gen LRU](#) è un moderno algoritmo di recupero delle pagine nel kernel Linux, progettato per migliorare le prestazioni di gestione della memoria in condizioni di pressione della memoria.

Sostituisce il tradizionale meccanismo LRU (Least Recently Used) utilizzato per determinare quali pagine di memoria recuperare quando il sistema sta esaurendo la memoria.

Il meccanismo LRU tradizionale utilizza un modello a due elenchi (attivo e inattivo) per tenere traccia dell'utilizzo delle pagine, il che può diventare inefficiente nei carichi di lavoro moderni con set di lavoro di grandi dimensioni. MGLRU lo sostituisce con più «generazioni» di pagine, permettendo al kernel di prendere decisioni più intelligenti sulla base di informazioni dettagliate sull'invecchiamento.

I vantaggi di MGLRU includono:

- Decisioni di recupero migliori: identificazione più accurata delle pagine fredde (non utilizzate).
- Latenza inferiore e velocità effettiva migliorata: in particolare per carichi di lavoro con ampi spazi di indirizzi o molti processi simultanei.
- Migliore conservazione della cache: le pagine utilizzate di recente hanno meno probabilità di essere rimosse prematuramente.
- Design scalabile ed efficiente in termini di blocco: offre prestazioni migliori su macchine con molte unità. CPUs

Configurazione e ottimizzazione

La configurazione del kernel `CONFIG_LRU_GEN` è abilitata sui AL2023 kernel. Questa viene compilata in MGLRU ma non la abilita di default.

MGLRU può essere abilitato e ottimizzato utilizzando il file `/sys/kernel/mm/lru_gen/enabled`. Il valore è una maschera di bit. Si consiglia di attivare tutti i componenti a meno che alcuni di essi non abbiano effetti collaterali indesiderati.

Bit	Componenti
0	Lo switch principale per la LRU multigenerazione.
1	Cancellazione del bit a cui si accede nelle voci della tabella Leaf Page in batch di grandi dimensioni, quando MMU lo imposta (ad esempio, su x86). Questo comportamento può teoricamente peggiorare lock contention (<code>mmap_lock</code>). Se è disabilitato, la LRU

Bit	Componenti
	multigenerazione subirà un lieve peggioramento delle prestazioni per i carichi di lavoro che mappano in modo contiguo le pagine calde, i cui bit a cui si accede potrebbero altrimenti essere cancellati da un minor numero di batch di dimensioni maggiori.
2	Cancellazione del bit a cui si accede anche nelle voci di tabella non a foglia, quando MMU lo imposta (ad esempio, su x86). Questo comportamento non è stato verificato su varietà x86 diverse da Intel e AMD. Se è disattivata, la LRU multigenerazione subirà un degrado delle prestazioni trascurabile.
[yYnN]	Enable/disable tutti i componenti di cui sopra.

Un esempio di come MGLRU può essere abilitato:

```
[ec2-user ~]$ echo y >/sys/kernel/mm/lru_gen/enabled
```

Ciò abilita tutti i componenti:

```
[ec2-user ~]$ cat /sys/kernel/mm/lru_gen/enabled
0x0007
```

Tutorial: Installazione LibreOffice su AL2023

LibreOffice è una suite di produttività per ufficio gratuita e open source che include applicazioni per l'elaborazione di testi, fogli di calcolo, presentazioni, diagrammi, database e modifica di formule. Puoi installarlo LibreOffice su istanze Amazon Linux 2023 (AL2023) per fornire agli utenti una suite desktop per ufficio completa.

LibreOffice è ufficialmente testato e supportato AL2023 da The Document Foundation. Per ulteriori informazioni, consulta lo stato di [avanzamento LibreOffice a 64 bit e il supporto per Amazon Linux 2023](#).

Indice

- [Prerequisiti](#)
- [Fase 1: Determinare l'architettura del sistema](#)
- [Fase 2: Scarica il pacchetto LibreOffice RPM](#)
- [Passaggio 3: Estrarre l'archivio](#)
- [Fase 4: Installare i pacchetti RPM](#)
- [Fase 5: Verificare l'installazione](#)
- [Avvio LibreOffice](#)
- [Risoluzione dei problemi](#)

Prerequisiti

- Un'istanza Amazon EC2 in esecuzione AL2023 con un ambiente desktop grafico GNOME configurato e accessibile tramite un protocollo desktop remoto (ad esempio, Amazon DCV). Per ulteriori informazioni, consulta gli argomenti seguenti:
 - [Tutorial: Installare l'ambiente desktop GNOME su AL2023](#)
 - [Configurazione del server Amazon DCV](#)
- sudo accesso root sull'istanza.
- Almeno 1,5 GB di spazio libero su disco per l'installazione.
- Una connessione Internet attiva per scaricare il LibreOffice pacchetto o il pacchetto prescaricato e trasferito sull'istanza.

Fase 1: Determinare l'architettura del sistema

Eseguite il comando seguente per identificare l'architettura del processore. Questo determina quale pacchetto RPM scaricare.

```
uname -p
```

L'output è uno dei seguenti:

Output	Architecture	Tipo di pacchetto RPM
x86_64	Intel/AMD a 64 bit	Linux_x86-64_rpm
aarch64	ARM a 64 bit (Graviton)	Linux_aarch64_rpm

Fase 2: Scarica il pacchetto LibreOffice RPM

Per scaricare il pacchetto LibreOffice RPM

1. Apri la [pagina di LibreOffice download](#).
2. Seleziona Linux (64 bit) (rpm) come sistema operativo.
3. Scegliete la versione appropriata per la vostra architettura (x86_64 o aarch64).
4. Scarica l'archivio sulla tua istanza .tar.gz. Puoi usare `wget` o `curl` scaricare direttamente dal terminale.

```
cd ~/Downloads
wget https://download.documentfoundation.org/libreoffice/stable/VERSION/
rpm/ARCHITECTURE/LibreOffice_VERSION_Linux_ARCHITECTURE_rpm.tar.gz
```

Sostituisci *VERSION* e *ARCHITECTURE* con il numero di versione e l'architettura che desideri installare (ad esempio, 25.2.5 e x86_64).

Note

[Per trovare il numero di versione stabile più recente, visita la pagina di download LibreOffice](#)

Passaggio 3: Estrarre l'archivio

Vai alla directory in cui hai scaricato l'archivio ed estrailo.

```
cd ~/Downloads
tar zxvf LibreOffice_VERSION_Linux_ARCHITECTURE_rpm.tar.gz
```

In questo modo viene creata una directory con un nome simile a `LibreOffice_`*VERSION*`_Linux_`*ARCHITECTURE*`_rpm/`.

Fase 4: Installare i pacchetti RPM

Per installare i pacchetti RPM

1. Passa alla `RPMS/` sottodirectory all'interno della cartella estratta.

```
cd LibreOffice_VERSION_Linux_ARCHITECTURE_rpm/RPMS/
```

2. Installa tutti i pacchetti RPM utilizzando `sudo rpm`

```
sudo rpm -i *.rpm
```

In alternativa, puoi usare `dnf` per la risoluzione delle dipendenze:

```
sudo dnf install -y *.rpm
```

Fase 5: Verificare l'installazione

Conferma che l'installazione LibreOffice è stata eseguita correttamente controllando la versione.

```
libreoffice --version
```

Output previsto:

```
LibreOffice VERSION ...
```

Avvio LibreOffice

Dal terminale

Esegui il comando seguente:

```
libreoffice
```

Se il `libreoffice` comando non è disponibile, è possibile creare una versione del file binario. Puoi invece avviarlo utilizzando il nome con versione:

```
libreofficeMAJOR_VERSION
```

Ad esempio, `libreoffice25.2`.

Per avviare direttamente un'applicazione specifica:

```
libreoffice --writer    # Word processor
libreoffice --calc      # Spreadsheet
libreoffice --impress   # Presentations
libreoffice --draw      # Diagrams
libreoffice --base      # Database
libreoffice --math      # Formula editor
```

Dal desktop GNOME

Dopo l'installazione, LibreOffice le applicazioni vengono visualizzate nel menu Attività di GNOME. Scegli Attività nell'angolo in alto a sinistra, cerca "LibreOffice" e seleziona l'applicazione desiderata.

Risoluzione dei problemi

La sezione seguente può aiutarti a risolvere i problemi più comuni durante l'installazione o l'esecuzione di LibreOffice AL2023

Avviso sull'ambiente Java Runtime

Alcune LibreOffice funzionalità (come Base e alcune estensioni) richiedono un Java Runtime Environment (JRE). Se vedi un avviso relativo alla mancanza di Java, installalo:

```
sudo dnf install -y java-17-amazon-corretto-headless
```

Quindi, LibreOffice accedi a Strumenti > Opzioni > LibreOffice> Avanzate e verifica che JRE sia stato rilevato.

LibreOffice comando non trovato

Se il `libreoffice` comando non viene trovato dopo l'installazione, è possibile che il file binario abbia una versione. Prova a eseguire:

```
libreofficeMAJOR_VERSION
```

Ad esempio, `libreoffice25.2`. Puoi anche individuare il file binario:

```
find /opt -name "soffice" 2>/dev/null
```

Risorse aggiuntive

Se riscontrate altri problemi, consultate la documentazione ufficiale di LibreOffice installazione per Linux:

- [LibreOffice Guida all'installazione di Linux](#)
- [LibreOffice Pagina di download](#)

Utilizzo di Amazon Linux 2023 al di fuori di Amazon EC2

Le immagini di container Amazon Linux 2023 possono essere eseguite in ambienti di runtime di container compatibili. Per ulteriori informazioni su come utilizzare Amazon Linux 2023 all'interno di un container, consulta [AL2023 in contenitori](#).

Amazon Linux 2023 (AL2023) può anche essere eseguito come guest virtualizzato al di fuori dell'esecuzione diretta su Amazon. EC2 Al momento ci sono KVM Sono qcow2 disponibili immagini VMware (OVA), () e Hyper-V (vhdx).

Note

La configurazione delle immagini di Amazon Linux 2023 è diversa da quella di Amazon Linux 2.

Se stai [utilizzando Amazon Linux 2 come macchina virtuale locale](#), dovrai adattare la configurazione per renderla compatibile con AL2 023.

Scarica immagini Amazon Linux 2023 da utilizzare con KVM e Hyper-V VMware

[Le immagini dei dischi di Amazon Linux 2023 da utilizzare con KVM e Hyper-V possono essere VMware scaricate da \[cdn.amazonlinux.com\]\(https://cdn.amazonlinux.com\).](#)

Configurazioni supportate di Amazon Linux 2023 per l'uso in ambienti virtualizzati non Amazon EC2

Questa sezione descrive i requisiti per l'esecuzione di Amazon Linux 2023 in ambienti EC2 virtualizzati non Amazon come KVM o Hyper-V. VMware

La base [AL2023 requisiti di sistema](#) si applica a tutti gli EC2 ambienti virtualizzati non Amazon. Un elenco dettagliato dei modelli di dispositivi supportati è riportato per ogni ambiente hypervisor nei seguenti argomenti.

KVM e Hyper-V offrono molte opzioni di configurazione e occorre prestare attenzione a configurarle per le esigenze di sicurezza, prestazioni e affidabilità. VMware Per ulteriori informazioni, consulta la documentazione fornita dall'hypervisor.

Argomenti

- [Requisiti per l'esecuzione AL2023 su KVM](#)
- [Requisiti per l'esecuzione di AL2 023 su VMware](#)
- [Requisiti per l'esecuzione di Amazon Linux 2023 su Hyper-V](#)

Requisiti per l'esecuzione AL2023 su KVM

Questa sezione descrive i requisiti per l'esecuzione AL2023 su KVM. Le immagini KVM di AL2023 sono disponibili per entrambe le architetture. `aarch64` `x86-64` Questi requisiti si aggiungono alla base [AL2023 requisiti di sistema](#) per le immagini KVM.

Argomenti

- [Requisiti dell'host KVM per l'esecuzione su KVM AL2023](#)
- [Supporto del dispositivo per KVM AL2023](#)
- [modalità di avvio \(UEFI/BIOS\) supporto per KVM AL2023](#)
- [Limitazioni in esecuzione AL2023 su KVM](#)

Requisiti dell'host KVM per l'esecuzione su KVM AL2023

Le immagini KVM sono attualmente qualificate su un host che esegue Ubuntu 22.04.3 LTS con la `qemu` versione `6.2+dfsg-2ubuntu6.15`, fornita da questa versione di Ubuntu, che utilizza un tipo di macchina per `e` un `q35` tipo di macchina per `x86-64` `virt` `aarch64`

Supporto del dispositivo per KVM AL2023

I modelli di **qemu** dispositivi testati per l'uso con immagini AL2023 KVM (sia che) sono **aarch64**:
x86-64

- `virtio-blk` (dispositivo a blocchi `virtio`)
- `virtio-scsi` (controller SCSI `virtio` con dispositivo disco)
- `virtio-net` (dispositivo di rete `virtio`)
- `ahci` (da utilizzare con l'unità CD-ROM virtuale)
- `usb-storage` (superiore a `xhci`)

I modelli di **qemu** dispositivi aggiuntivi abilitati alla qualificazione delle immagini AL2023 KVM, ma non particolarmente utilizzati, sono:

- VGA (qemu VGA) solo su x86-64
- `virtio-rng` (generatore virtuale di numeri casuali)
- Dispositivi tastiera AT e mouse PS/2 legacy
- Dispositivo seriale legacy

modalità di avvio (UEFI/BIOS) supporto per KVM AL2023

L'immagine x86-64 viene testata sia con entrambe le modalità di avvio legacy, BIOS e UEFI. Le immagini aarch64 vengono testate con la modalità di avvio UEFI.

Note

Per impostazione predefinita, quando si utilizza la modalità di UEFI avvio, alcuni gestori di macchine virtuali forniranno alla macchina virtuale le chiavi Microsoft Secure Boot che abilitano Secure Boot. Questa configurazione non si avvia AL2023.

Poiché il AL2023 boot loader non è firmato da Microsoft, è necessario effettuare il provisioning della macchina virtuale senza chiavi UEFI o con le AL2023 chiavi per Secure Boot.

Important

Il supporto Secure Boot per KVM le immagini non è stato ancora convalidato.

Limitazioni in esecuzione AL2023 su KVM

Esistono alcune limitazioni note nell'esecuzione AL2023 su KVM.

Note

Il codice che implementa alcune delle funzionalità non supportate elencate potrebbe esistere AL2023 e funzionare correttamente. L'elenco delle funzionalità non supportate è disponibile

in modo da consentirti di prendere decisioni informate su cosa fare affidamento oggi e su ciò che il team di Amazon Linux considererà funzionante nell'ambito dei futuri aggiornamenti.

Limitazioni note relative all'esecuzione AL2023 su KVM

- L'agente guest KVM non è attualmente incluso nel pacchetto né supportato.
- Non sono supportati il collegamento e lo scollegamento a caldo di CPU, memoria o qualsiasi altro tipo di dispositivo.
- L'ibernazione delle VM non è supportata.
- La migrazione delle macchine virtuali non è supportata.
- Il passthrough di qualsiasi dispositivo, ad esempio PCI Passthrough o USB Passthrough, non è supportato.

Requisiti per l'esecuzione di AL2 023 su VMware

Questa sezione descrive i requisiti per eseguire AL2 023 su VMware. La VMware le immagini di AL2 023 sono disponibili solo per l'x86-64architettura. VMware le immagini per non aarch64 sono disponibili o supportate. Questi requisiti si aggiungono alla base [AL2023 requisiti di sistema](#) per VMware immagini.

Argomenti

- [VMware requisiti dell'host per l'esecuzione di AL2 023 su VMware](#)
- [Supporto per dispositivi dalla versione AL2 023 in poi VMware](#)
- [modalità di avvio \(UEFI e BIOS\) supporto per AL2 023 in poi VMware](#)
- [Limitazioni in esecuzione su 023 AL2 VMware](#)

VMware requisiti dell'host per l'esecuzione di AL2 023 su VMware

Il AL2 023 VMware Le immagini OVA sono attualmente qualificate per quanto segue:

- VMware Workstation 17.5.0 in esecuzione su host che utilizzano un processore Intel (R) Xeon (R) Platinum 8124M
- VMware vSphere 8.0 con un processore Intel (R) Xeon (R) Platinum 8275CL

AL2023 VMWare Le immagini OVA specificano una versione hardware della macchina pari a 13.

VMWare La versione 13 dell'hardware della macchina è supportata da:

- ESXi 6.5 o versione successiva
- VMWare Workstation 14 o versione successiva

Supporto per dispositivi dalla versione AL2 023 in poi VMWare

I seguenti VMWare i modelli di dispositivi sono stati testati per l'uso con AL2 023 VMWare Immagini OVA (**x86-64**solo):

- `vmw_pvscsi` (VMWare paravirtualizzate SCSI controllore)
- `vmxnet3` (VMWare dispositivo di rete paravirtualizzato)
- `ata_piix`(eredità IDE da utilizzare solo con l'unità CD-ROM virtuale)

Aggiuntivo VMWare modelli di dispositivi abilitati nel AL2 2023 VMWare qualificazione dell'immagine, ma non esercitata in modo intensivo:

- `vmw_vmci` relativa `vsock` interfaccia (trasporto tramite socket virtuale per VMWare agente ospite)
- Dispositivo di memoria balloon `vmw_balloon`
- VMWare SVGAcontrollore
- legacy AT tastiera e PS/2 dispositivi mouse

Il VMWare il pacchetto guest agent (`open-vm-tools`) è disponibile e installato per impostazione predefinita nella versione AL2 023 VMWare immagini OVA.

modalità di avvio (UEFI e BIOS) supporto per AL2 023 in poi VMWare

A partire dalla versione 2023.3.20231211, la 023 AL2 VMWare L'immagine OVA è stata convalidata in entrambe le versioni precedenti BIOS e UEFI modalità di avvio. La configurazione predefinita di OVA è ancora obsoleta BIOS ma può essere modificata dall'utente.

Important

Il supporto Secure Boot richiede UEFI, che non è stato convalidato per AL2 023 in esecuzione su VMWare.

Limitazioni in esecuzione su 023 AL2 VMware

Esistono alcune limitazioni note nell'esecuzione di AL2 023 su VMware.

Note

Il codice che implementa alcune delle funzionalità non supportate elencate può esistere in AL2 023 e funzionare correttamente. L'elenco delle funzionalità non supportate consente ai clienti di prendere decisioni informate su cosa poter utilizzare attualmente per lavorare, nonché su quello che il team di Amazon Linux renderà disponibile nell'ambito dei futuri aggiornamenti.

Limitazioni note relative all'esecuzione di 023 su AL2 VMware

- UEFI Secure Boot non è attualmente convalidato con 023 attivo AL2 VMware.
- Non sono supportati il collegamento e lo scollegamento a caldo di CPU, memoria o qualsiasi altro tipo di dispositivo.
- L'ibernazione delle VM non è supportata.
- La migrazione delle macchine virtuali non è supportata.
- Il passthrough di qualsiasi dispositivo, ad esempio PCI Passthrough o USB Passthrough, non è supportato.

Requisiti per l'esecuzione di Amazon Linux 2023 su Hyper-V

Questa sezione descrive i requisiti per eseguire Amazon Linux 2023 su Hyper-V. Le immagini Hyper-V della versione AL2 023 sono disponibili solo per l'architettura. x86-64 Le immagini Hyper-V per non aarch64 sono attualmente disponibili o supportate.

Questa sezione descrive i requisiti aggiuntivi oltre alla base [AL2023 requisiti di sistema](#) per le immagini Hyper-V.

Argomenti

- [Requisiti dell'host Hyper-V per l'esecuzione di Amazon Linux 2023 su Hyper-V](#)
- [Supporto per dispositivi per Amazon Linux 2023 su Hyper-V](#)
- [Limitazioni all'esecuzione di Amazon Linux 2023 su Hyper-V](#)

Requisiti dell'host Hyper-V per l'esecuzione di Amazon Linux 2023 su Hyper-V

La qualificazione principale di Amazon Linux 2023 su Hyper-V avviene su Windows Server 2022 in esecuzione su un'istanza. EC2 c5.metal

Supporto per dispositivi per Amazon Linux 2023 su Hyper-V

Amazon Linux 2023 è testato su macchine virtuali Hyper-V di prima e seconda generazione con il seguente set di hardware virtualizzato:

- VM di prima generazione (avvio BIOS legacy)
- VM di seconda generazione (avvio UEFI - nessun avvio sicuro)
- I seguenti modelli di dispositivi sono testati per l'uso con AL2 023 immagini Hyper-V:
 - Storage virtuale Hyper-V *hv_storvsc* per il disco principale e l'unità CD-ROM emulata di seconda generazione VMs
 - IDE PIIX emulato *ata_piix* per l'unità CD-ROM virtuale di prima generazione VMs
 - Ethernet virtuale Hyper-V *hv_netvsc*
- I seguenti modelli di dispositivi sono abilitati ma leggermente testati:
 - Modalità di testo VGA legacy sulla prima generazione VMs
 - *simpledmmfbFramebuffer* basato su firmware UEFI di seconda generazione VMs
 - Palloncino Hyper-V *hv_balloon*
 - Hyper-V HID/Mouse *hid_hyperv*
- Le seguenti modalità del dispositivo non sono attualmente abilitate in AL2 023:
 - Pass-through PCI Hyper-V
 - Grafica DRM Hyper-V

Important

Per le macchine virtuali di seconda generazione, Secure Boot non è supportato e deve essere disabilitato prima di avviare la macchina virtuale per avviare correttamente Amazon Linux 2023. Hyper-V attualmente supporta solo Secure Boot con componenti software firmati con chiavi proprie di Microsoft, mentre il bootloader Amazon Linux è firmato da una chiave privata Amazon. Hyper-V non supporta l'importazione di chiavi di terze parti a questo punto.

Limitazioni all'esecuzione di Amazon Linux 2023 su Hyper-V

Di seguito sono riportate alcune limitazioni note nell'esecuzione di Amazon Linux 2023 su Hyper-V:

Note

Il codice che implementa alcune delle funzionalità non supportate elencate potrebbe esistere in AL2 023 e funzionare correttamente. L'elenco delle funzionalità non supportate consente ai clienti di prendere decisioni informate su cosa poter utilizzare attualmente per lavorare, nonché su quello che il team di Amazon Linux renderà disponibile nell'ambito dei futuri aggiornamenti.

Limitazioni note relative all'esecuzione di AL2 023 su Hyper-V

- La modalità UEFI Secure Boot non è attualmente supportata né funzionante con 023 su Hyper-V AL2
- Non sono supportati il collegamento e lo scollegamento a caldo di CPU, memoria o qualsiasi altro tipo di dispositivo.
- L'ibernazione delle macchine virtuali (VM) non è supportata.
- La migrazione delle macchine virtuali (VM) non è supportata.
- Il passthrough di qualsiasi dispositivo, ad esempio PCI Passthrough o USB Passthrough, non è supportato.

Amazon Linux 2023 **cloud-init** Configurazione e configurazione se utilizzato al di fuori di Amazon EC2

Questa sezione spiega come configurare una macchina virtuale Amazon Linux 2023 quando non viene eseguita direttamente su Amazon EC2, ad esempio su KVM o Hyper-V. VMware

Per impostazione predefinita, le immagini di una macchina virtuale Amazon Linux 2023 non vengono fornite con alcuna password utente o chiave ssh e ottengono la configurazione di rete tramite DHCP sulla prima interfaccia di rete scoperta. Ciò significa che, senza configurazioni aggiuntive, non è possibile connettersi alla macchina virtuale risultante per impostazione predefinita.

Pertanto, è necessario fornire una qualche forma di configurazione per la macchina virtuale. Il meccanismo standard per eseguire questa operazione per Amazon Linux è tramite origini dati `cloud-init`.

Amazon Linux 2023 è stato qualificato con le seguenti origini dati:

NoCloud

Questo è il metodo tradizionale di configurazione delle immagini locali tramite un CD-ROM virtuale contenente un seed ISO9660 immagine con file di configurazione. `cloud-init`

VMware

Amazon Linux 2023 supporta inoltre la configurazione di VMware immagini in esecuzione su vSphere tramite VMware l'origine `VMware guestinfo.userdata` dati specifica tramite e. `guestinfo.metadata`

Note

La configurazione delle origini dati può essere diversa rispetto a quella di Amazon Linux 2. Più specificamente, Amazon Linux 2023 usa `systemd-networkd` per la sua configurazione e richiede l'uso di `cloud-init` "Configurazione di rete versione 2", come descritto nella [documentazione sulla configurazione di rete di cloud-init](#).

La documentazione completa per i meccanismi di configurazione `cloud-init` per la versione di `cloud-init` pacchettizzata in Amazon Linux 2023 è disponibile nella [documentazione sulla versione upstream di cloud-init](#).

NoCloud (**seed.iso**) **cloud-init** configurazione per Amazon Linux 2023 su KVM e VMware

Questa sezione spiega come creare e utilizzare un `seed.iso` immagine per configurare Amazon Linux 2023 in esecuzione su KVM oppure VMware. Perché KVM e VMware gli ambienti non dispongono di [Amazon EC2 Instance Meta Data Service \(IMDS\)](#), è richiesto un metodo alternativo per configurare Amazon Linux 2023 e fornire `seed.iso` un'immagine è uno di questi metodi.

L'immagine di avvio `seed.iso` include le informazioni di configurazione iniziale necessarie per avviare e configurare la nuova VM, ad esempio la configurazione di rete, il nome host e i dati utente.

Note

L'immagine `seed.iso` include solo le informazioni di configurazione richieste per avviare la VM. Non include invece i file del sistema operativo Amazon Linux 2023.

Per generare l'immagine `seed.iso`, sono necessari almeno due file di configurazione, talvolta tre:

meta-data

Questo file solitamente include il nome host per la macchina virtuale.

user-data

Questo file configura in genere gli account utente, le relative password, ssh coppie di chiavi e/o meccanismi di accesso. Per impostazione predefinita, il KVM e VMware le immagini di Amazon Linux 2023 creano un account `ec2-user` utente. Puoi usare il file di configurazione `user-data` per impostare la password e/o le chiavi ssh per tale account utente predefinito.

network-config(opzionale)

Questo file fornisce solitamente una configurazione di rete per la macchina virtuale che va a sostituire quella predefinita. La configurazione predefinita è quella di utilizzare DHCP sulla prima interfaccia di rete disponibile.

Creazione dell'immagine disco **seed.iso**

1. Su un computer Linux o macOS, puoi creare una nuova cartella denominata `seedconfig` ed esplorarne il contenuto.

Note

È possibile usare Windows o un altro sistema operativo per completare questi passaggi, ma è necessario trovare uno strumento equivalente a `mkisofs` per completare la creazione dell'immagine `seed.iso`.

2. Crea il file di configurazione `meta-data`.
 - a. Crea un nuovo file denominato `meta-data`.

- b. Apri il meta-data file utilizzando il tuo editor preferito e aggiungi quanto segue, sostituendolo *vm-hostname* con il nome host della macchina virtuale:

```
#cloud-config
local-hostname: vm-hostname
```

- c. Salva e chiudi il file di configurazione meta-data.
3. Crea il file di configurazione user-data.
 - a. Crea un nuovo file denominato `user-data`.
 - b. Apri il file `user-data` utilizzando l'editor preferito e aggiungi quanto segue, effettuando le opportune sostituzioni:

```
#cloud-config
#vim:syntax=yaml
users:
# A user by the name 'ec2-user' is created in the image by default.
- default
- name: ec2-user
ssh_authorized_keys:
- ssh-rsa ssh-key
# In the above line, replace ssh key with the content of your ssh public key.
```

- c. Facoltativamente, puoi aggiungere altri account utente al `user-data` file di configurazione.

Puoi specificare account utente aggiuntivi, definendone i meccanismi di accesso, le password e le coppie di chiavi. Per ulteriori informazioni sulle direttive supportate, consulta la [documentazione della versione upstream di cloud-init](#).

- d. Salva e chiudi il file di configurazione `user-data`.
4. (Facoltativo) Crea il file di configurazione `network-config`.
 - a. Crea un nuovo file denominato `network-config`.
 - b. Apri il file `network-config` utilizzando l'editor preferito e aggiungi quanto segue, sostituendo i vari indirizzi IP con quelli appropriati per la configurazione desiderata.

```
#cloud-config
version: 2
ethernets:
```

```
enp1s0:
  addresses:
    - 192.168.122.161/24
  gateway4: 192.168.122.1
  nameservers:
    addresses: 192.168.122.1
```

Note

`cloud-init` la configurazione di rete fornisce meccanismi di confronto con MAC indirizzo dell'interfaccia invece di specificare il nome dell'interfaccia che può cambiare a seconda della configurazione della macchina virtuale. Queste (e altre) funzionalità `cloud-init` per la configurazione di rete sono descritte più dettagliatamente nella [documentazione della versione 2 della configurazione di rete upstream cloud-init](#).

- c. Salva e chiudi il file di configurazione `network-config`.
5. Crea l'immagine disco `seed.iso` utilizzando i file di configurazione `meta-data`, `user-data` e `network-config` opzionale che sono stati creati nei passaggi precedenti.

Esegui una delle seguenti operazioni, a seconda del sistema operativo su stai creando l'immagine disco `seed.iso`.

- Sui sistemi Linux, usa uno strumento come **`mkisofs`** o **`genisoimage`** per creare il file `seed.iso` completo. Vai alla cartella `seedconfig` ed esegui il comando seguente:

```
$ mkisofs -output seed.iso -volid cidata -joliet -rock user-data meta-data
```

- Se usi una `network-config`, includila nell'invocazione di **`mkisofs`**:

```
$ mkisofs -output seed.iso -volid cidata -joliet -rock user-data meta-data
network-config
```

- Sui sistemi macOS, puoi usare uno strumento come **`hdiutil`** per generare il file `seed.iso` completo. Poiché **`hdiutil`** richiede un nome di percorso anziché un elenco di file, la stessa invocazione può essere utilizzata indipendentemente dal fatto che un file di configurazione `network-config` sia stato creato o meno.

```
$ hdiutil makehybrid -o seed.iso -hfs -joliet -iso -default-volume-name cidata seedconfig/
```

6. Il `seed.iso` file risultante può ora essere allegato alla tua nuova macchina virtuale Amazon Linux 2023 utilizzando un'unità CD-ROM virtuale da trovare `cloud-init` al primo avvio e applicare la configurazione al sistema.

VMwarecloud-initconfigurazione guestinfo per on AL2023 VMware

VMware gli ambienti non dispongono di [Amazon EC2 Instance Meta Data Service \(IMDS\)](#), quindi è necessario un metodo di configurazione AL2023 alternativo. Questa sezione descrive come utilizzare un meccanismo di configurazione alternativo all'unità CD-ROM `seed.iso` virtuale disponibile in VMware vSphere.

Questo metodo di configurazione utilizza il VMware `extraconfig` meccanismo per fornire i dati di configurazione a `cloud-init`. Per ciascuna delle seguenti chiavi, deve essere fornita una **keyname.encoding** proprietà corrispondente.

È possibile fornire al VMware `extraconfig` meccanismo le seguenti chiavi.

guestinfo.metadata

JSON o YAML contenenti metadati `cloud-init`

guestinfo.userdata

Un documento YAML contenente dati utente `cloud-init` nel formato `cloud-config`.

guestinfo.vendordata (facoltativo)

YAML contenente i dati del `cloud-init` fornitore

Le proprietà di codifica corrispondenti (`guestinfo.metadata.encoding`, `guestinfo.userdata.encoding` e `guestinfo.vendordata.encoding`) possono contenere:

base64

Il contenuto della proprietà è codificato con `base64`.

gzip+base64

Il contenuto della proprietà è compresso con `gzip` dopo la codifica con `base64`.

Note

Il `seed.iso` metodo supporta un file di `network-config` configurazione separato (opzionale). `VMwareguestinfo` differisce nel modo in cui viene fornita la configurazione di rete. Ulteriori informazioni sono fornite nella sezione seguente.

Se desideri una configurazione di rete esplicita, devi incorporarla in `metadata` a sotto forma di due proprietà YAML o JSON:

network

Contiene la configurazione di rete codificata in formato JSON o YAML.

network.encoding

Contiene la codifica dei suddetti dati di configurazione di rete. Le codifiche supportate da `cloud-init` sono le stesse dei dati `guestinfo`: `base64` e `gzip+base64`.

Example Utilizzo dello strumento VMware vSphere **govc** CLI per passare la configurazione con **guestinfo**

1. Preparare `meta-data` i `user-data` file di `network-config` configurazione e quelli opzionali come descritto in [NoCloud \(seed.iso\) cloud-init configurazione per Amazon Linux 2023 su KVM e VMware](#)
2. Convertire i file di configurazione in formati utilizzabili da `VMwareguestinfo`.

```
# 'meta-data', `user-data` and `network-config` are the configuration
# files in the same format that would be used by a NoCloud (seed.iso)
# data source, read-them and convert them to VMware guestinfo
#
# The VM_NAME variable is assumed to be set to the name of the VM
# It is assumed that the necessary govc environment (credentials etc...) are
# already set

metadata=$(cat "meta-data")
userdata=$(cat "user-data")
if [ -e "network-config" ] ; then
    # We need to embed the network config inside the meta-data
    netconf=$(base64 -w0 "network-config")
```

```

    metadata=$(printf "%s\nnetwork: %s\nnetwork.encoding: base64" "$metadata"
"$netconf")
fi
metadata=$(base64 -w0 <<< "$metadata")
govc vm.change -vm "$VM_NAME" \
    -e guestinfo.metadata="$metadata" \
    -e guestinfo.metadata.encoding="base64"
userdata=$(base64 -w0 <<< "$userdata")
govc vm.change -vm "$VM_NAME" \
    -e guestinfo.userdata="$userdata" \
    -e guestinfo.userdata.encoding="base64"

```

Confronto dei pacchetti installati sull'AMI standard di Amazon Linux 2023 con l'immagine AL2 KVM 023

Un confronto tra la versione RPMs attuale dell'AMI standard AL2 023 e quella RPMs presente sull'immagine KVM AL2 023.

Pacchetto	AMI	KVM
acl	2.3.1	2.3.1
acpid	2.0.32	
alternatives	1.15	1.15
amazon-chroney-config	4.3	
amazon-ec2-net-utils	2.5.1	
amazon-linux-onprem		1.2
amazon-linux-repo-cdn		2023,620241031
amazon-linux-repo-s3	2023,620241031	
amazon-linux-sb-keys	2023,1	2023,1

Pacchetto	AMI	KVM
amazon-onprem-network		1.2
amazon-rpm-config	228	228
amazon-ssm-agent	3,3,987,0	3,3,987,0
amd-ucode-firmware	20210208 (marzo)	20210208 (novembre)
at	3.1.23	3,1,23
attr	2.5.1	2.5.1
audit	30,6	3.0.6
audit-libs	3.0.6	3.0.6
aws-cfn-bootstrap	2.0	
awscli-2	2,15,30	2,15,30
basesystem	11	11
bash	5,2,15	5,2,15
bash-completion	2.11	2.11
bc	1,07,1	1,07,1
bind-libs	9,18,28	9,18,28
bind-license	9,18,28	9,18,28
bind-utils	9,18,28	9,18,28
binutils	2,39	2,39
boost-filesystem	1,75,0	1,75,0
boost-system	1,75,0	1,75,0

Pacchetto	AMI	KVM
boost-thread	1,75,0	1,75,0
bzip2	1.0.8	1.0.8
bzip2-libs	1.0.8	1.0.8
ca-certificates	2023,2,68	2023,2,68
c-ares	1.19.1	
checkpolicy	3.4	3.4
chkconfig	1.15	1.15
chrony	4.3	4.3
cloud-init	222,2	222,2
cloud-init-cfg-ec2	222,2	
cloud-init-cfg-onpre		222,2
cloud-utils-growpart	0,31	0,31
coreutils	8,32	8,32
coreutils-common	8,32	8,32
cpio	2,13	2,13
cracklib	2,9,6	29.6
cracklib-dicts	29.6	29.6
crontabs	1.11	1.11
crypto-policies	20220428	20220428

Pacchetto	AMI	KVM
crypto-policies-scripts	20220428	20220428
cryptsetup	2.6.1	2.6.1
cryptsetup-libs	2.6.1	2.6.1
curl-minimal	8,5,0	8,5,0
cyrus-sasl-lib	2,1,27	2,1,27
cyrus-sasl-plain	2,1,27	2,1,27
dbus	1,12,28	1,12,28
dbus-broker	32	32
dbus-common	1,12,28	1,12,28
dbus-libs	1,12,28	1,12,28
device-mapper	1,02,185	1,02,185
device-mapper-libs	1,02,185	1,02,185
diffutils	3.8	3.8
dnf	4.14.0	4.14.0
dnf-data	4.14.0	4.14.0
dnf-plugin-release-notification	1.2	1.2
dnf-plugins-core	4.3.0	4.3.0
dnf-plugin-support-info	1.2	1.2

Pacchetto	AMI	KVM
dnf-utils	4.3.0	4.3.0
dosfstools	4.2	4.2
dracut	055	055
dracut-config-ec2	3.0	
dracut-config-generic	055	055
dwz	0,14	0,14
dyninst	102,1	10.2.1
e2fsprogs	1,446,5	1,46,5
e2fsprogs-libs	1,46,5	1,46,5
ec2-hibinit-agent	1.0.8	
ec2-instance-connect	1.1	
ec2-instance-connect-selinux	1.1	
ec2-utils	2.2.0	
ed	1.14.2	1.14.2
efi-filesystem	5	5
efi-srpm-macros	5	5
efivar	38	38
efivar-libs	38	38

Pacchetto	AMI	KVM
elfutils-debuginfod-client	0.188	0.188
elfutils-default-yama-scope	0.188	0.188
elfutils-libelf	0.188	0.188
elfutils-libs	0.188	0.188
ethtool	5,15	5,15
expat	2.5.0	2.5.0
file	5,39	5,39
file-libs	5,39	5,39
filesystem	3,14	3,14
findutils	4.8.0	4.8.0
fonts-srpm-macros	2.0.5	2.0.5
fstrm	0.6.1	0.6.1
fuse-libs	2,9,9	29.9
gawk	5.1.0	5.1.0
gdbm-libs	1,19	1,19
gdisk	1.0.8	1.0.8
gettext	0,21	0,21
gettext-libs	0,21	0,21
ghc-srpm-macros	1.5.0	1.5.0

Pacchetto	AMI	KVM
glib2	2,74,7	2,74,7
glibc	2,34	2,34
glibc-all-langpacks	2,34	2,34
glibc-common	2,34	2,34
glibc-gconv-extra	2,34	2,34
glibc-locale-source	2,34	2,34
gmp	62,1	6.2.1
gnupg2-minimal	2.3.7	2.3.7
gnutls	3.8.0	3.8.0
go-srpm-macros	3.2.0	3.2.0
gpgme	1.15.1	1.15.1
gpm-libs	1,20.7	1,20.7
grep	3.8	3.8
groff-base	1,22,4	1,22,4
grub2-common	2,06	2,06
grub2-efi-aa64-ec2	2,06 (aarch64)	2.06 (aarch64)
grub2-efi-x64-ec2	2,06 (x86_64)	2,06 (x86_64)
grub2-pc		2,06 (x86_64)
grub2-pc-modules	2,06	2.06 (nomarzo)
grub2-tools	2.06	2,06

Pacchetto	AMI	KVM
grub2-tools-minimal	2,06	2,06
grubby	8,40	8,40
gssproxy	0.8.4	0.8.4
gzip	1.12	1.12
hostname	3,23	3,23
hunspell	1.7.0	1.7.0
hunspell-en	0,20140811,1	0,20140811,1
hunspell-en-GB	0,20140811,1	0,20140811,1
hunspell-en-US	0,20140811,1	0,20140811,1
hunspell-filesystem	1.7.0	1.7.0
hwdata	0,384	0,384
info	6.7	6.7
inih	49	49
initscripts	10,09	10,09
iproute	6.10.0	6.10.0
iputils	20210202	20210202
irqbalance	1.9.0	1.9.0
jansson	2.14	2.14
jemalloc	52,1	5.2.1
jitterentropy	3.4.1	3.4.1

Pacchetto	AMI	KVM
jq	1.7.1	
json-c	0,14	0,14
kbd	2.4.0	2.4.0
kbd-misc	2.4.0	2.4.0
kernel	6,1112	6,1112
kernel-libbpf	6,1112	6,1112
kernel-livepatch-r epo-cdn		2023,620241031
kernel-livepatch-r epo-s3	2023,620241031	
kernel-modules-extra		6,1112
kernel-modules-ext ra-common		6,1112
kernel-srpm-macros	1.0	1.0
kernel-tools	6,1112	6,1112
keyutils	1.6.3	1.6.3
keyutils-libs	1.6.3	1.6.3
kmod	29	29
kmod-libs	29	29
kpatch-runtime	0,9,7	0,9,7
krb5-libs	1,21,3	1,21,3

Pacchetto	AMI	KVM
less	608	608
libacl	2.3.1	2.3.1
libaio	0,3111	0,3111
libarchive	3,7,4	3,7,4
libargon2	20171227	20171227
libassuan	2,5,5	2,5,5
libattr	2.5.1	2.5.1
libbasicobjects	0,11	01.1
libblkid	2,37,4	2,37,4
libcap	2,48	2,48
libcap-ng	0.8.2	0.8.2
libcbor	0.7.0	0.7.0
libcollection	0.7.0	0.7.0
libcom_err	1,446,5	1,46,5
libcomps	01,20	01,20
libconfig	17.2	1.7.2
libcurl-minimal	8,5,0	8,5,0
libdb	5,3,28	5,3,28
libdhash	0,5,0	
libdnf	0,69,0	0,69,0

Pacchetto	AMI	KVM
libeconf	0,40	0,40
libedit	3.1	3.1
libev	4,33	4,33
libevent	2,1,12	2,1,12
libfdisk	2,37,4	2,37,4
libffi	34.4	34.4
libfido2	1.10.0	1.10.0
libgcc	114,1	11.4.1
libgcrypt	1.10.2	1.10.2
libgomp	11.4.1	11.4.1
libgpg-error	1,42	1,42
libibverbs	48,0	48,0
libidn2	2.3.2	2.3.2
libini_config	1.3.1	1.3.1
libkcapi	1.4.0	1.4.0
libkcapi-hmacalc	1.4.0	1.4.0
libldb	26.2	
libmaxminddb	1.5.2	1.5.2
libmetalink	0,13	0,13
libmnl	1.0.4	1.0.4

Pacchetto	AMI	KVM
libmodulemd	2.13.0	2.13.0
libmount	2,37,4	2,37,4
libnfsidmap	2,5,4	2.5.4
libnghttp2	1,59,0	1,59,0
libnl3	3.5.0	3.5.0
libpath_utils	0,2,1	0,2,1
libpcap	1.10.1	1.10.1
libpipeline	1.5.3	1.5.3
libpkgconf	1.8.0	1.8.0
libpsl	0,21,1	0,21,1
libpwquality	1.4.4	1.4.4
libref_array	0,1,5	0,1,5
librepo	1,14,5	1,14,5
libreport-filesystem	2,15,2	2,15,2
libseccomp	2.5.3	2.5.3
libselinux	3.4	3.4
libselinux-utils	3.4	3.4
libsemanage	3.4	3.4
libsepol	3.4	3.4
libsigsegv	2,13	2,13

Pacchetto	AMI	KVM
libsmartcols	2,37,4	2,37,4
libsolv	0,7,22	0,7,22
libss	1,446,5	1,46,5
libsss_certmap	2,9,4	
libsss_idmap	2,9,4	2,9,4
libsss_nss_idmap	2,9,4	2,9,4
libsss_sudo	2,9,4	
libstdc++	11.4.1	11.4.1
libstoragegmt	1.9.4	1.9.4
libtalloc	2.3.4	
libtasn1	4,19,0	4,19,0
libtdb	1.4.7	
libtevent	0.13.0	
libtextstyle	0,21	0,21
libtirpc	1.3.3	1.3.3
libunistring	0,9,10	0,9,10
libuser	0,63	0,63
libutempter	1.2.1	1.2.1
libuuid	2,37,4	2,37,4
libuv	1.47.0	1.47.0

Pacchetto	AMI	KVM
libverto	0,32	0,32
libverto-libev	0,32	0,32
libxcrypt	4,4,33	4,4,33
libxml2	2,10,4	210.4
libyaml	0,2,5	02,5
libzstd	1,5,5	1,5,5
linux-firmware-whe nce	20210208 (novembre)	20210208 (novembre)
lm_sensors-libs	3.6.0	3.6.0
lmdb-libs	0,9,29	0,9,29
logrotate	3,20,1	3,20,1
lsyf	4,94,0	4,94,0
lua-libs	5.4.4	5.4.4
lua-srpm-macros	1	1
lz4-libs	1.9.4	1.9.4
man-db	2,9,3	29.3
man-pages	5,10	5,10
microcode_ctl	2.1 (x86_64)	2.1 (x86_64)
mpfr	4.1.0	4.1.0
nano	5.8	5.8
ncurses	6.2	6.2

Pacchetto	AMI	KVM
ncurses-base	6.2	6.2
ncurses-libs	6.2	6.2
nettle	3.8	3.8
net-tools	2.0	2.0
newt	0,52,21	0,52,21
nfs-utils	2,5,4	2.5.4
npth	1.6	1.6
nspr	4,35,0	4,35,0
nss	3,90,0	3,90,0
nss-softokn	3,90,0	3,90,0
nss-softokn-freebl	3,90,0	3,90,0
nss-sysinit	3,90,0	3,90,0
nss-util	3,90,0	3,90,0
ntsysv	1.15	1.15
numactl-libs	2.0,14	2.0,14
ocaml-srpm-macros	6	6
oniguruma	6,9,7,1	
openblas-srpm-macros	2	2
openldap	2,4,57	2,4,57
openssh	8.7p1	8,7p1

Pacchetto	AMI	KVM
openssh-clients	8,7p1	8,7p1
openssh-server	8,7p1	8,7p1
openssl	3,0,8	30,8
openssl-libs	30,8	30,8
openssl-pkcs11	0,4,12	0,4,12
os-prober	1,77	1,77
p11-kit	0,24,1	0,24,1
p11-kit-trust	0,24,1	0,24,1
package-notes-srpm-macros	0.4	0.4
pam	1.5.1	1.5.1
parted	3.4	3.4
passwd	0,80	0,80
pciutils	3.7.0	3.7.0
pciutils-libs	3.7.0	3.7.0
pcre2	10,40	10,40
pcre2-syntax	10,40	10,40
perl-Carp	1,50	1,50
perl-Class-Struct	0,66	0,66
perl-constant	1,33	1,33
perl-DynaLoader	1,47	1,47

Pacchetto	AMI	KVM
perl-Encode	3,15	3,15
perl-Errno	1,30	1,30
perl-Exporter	5,74	5,74
perl-Fcntl	1.13	1.13
perl-File-Basename	2,85	2,85
perl-File-Path	2,18	2,18
perl-File-stat	1,09	1,09
perl-File-Temp	0,231,100	0,231,100
perl-Getopt-Long	2,52	2,52
perl-Getopt-Std	1.12	1.12
perl-HTTP-Tiny	0,078	0,078
perl-if	0,60,800	0,60,800
perl-interpreter	5,32,1	5,32,1
perl-IO	1,43	1,43
perl-IPC-Open3	1,21	1,21
perl-libs	5,32,1	5,32,1
perl-MIME-Base64	3,16	3,16
perl-mro	1,23	1,23
perl-overload	1,31	1,31
perl-overloading	0,02	0,02

Pacchetto	AMI	KVM
perl-parent	0,238	0,238
perl-PathTools	3,78	3,78
perl-Pod-Escapes	1,07	1,07
perl-podlators	4,14	4,14
perl-Pod-Perldoc	3,28,01	3,28,01
perl-Pod-Simple	3,42	3,42
perl-Pod-Usage	2,01	2,01
perl-POSIX	1,94	1,94
perl-Scalar-List-Utils	1,56	1,56
perl-SelectSaver	1.02	1.02
perl-Socket	2,032	2,032
perl-srpm-macros	1	1
perl-Storable	3,21	3,21
perl-subst	1,03	1,03
perl-Symbol	1,08	1,08
perl-Term-ANSIColor	5,01	5,01
perl-Term-Cap	1,17	1,17
perl-Text-ParseWords	3,30	3,30
perl-Text-Tabs+Wrap	2021,0726	2021,0726
perl-Time-Local	1,300	1.300

Pacchetto	AMI	KVM
perl-vars	1,05	1,05
pkgconf	1.8.0	1.8.0
pkgconf-m4	1.8.0	1.8.0
pkgconf-pkg-config	1.8.0	1.8.0
policycoreutils	3.4	3.4
policycoreutils-python-utils	3.4	
popt	1,18	1,18
procps-ng	3,3,17	3,3,17
protobuf-c	1.4.1	1.4.1
psacct	6,6,4	6.6.4
psmisc	23,4	23,4
publicsuffix-list-dafsa	20240212	20240212
python3	3,9,16	3,9,16
python3-attrs	20,3,0	20,3,0
python3-audit	30.6	3.0.6
python3-awscrt	0,19,19	0,19,19
python3-babel	2,9,1	29.1
python3-cffi	1,14,5	1,14,5
python3-chardet	4.0.0	4.0.0

Pacchetto	AMI	KVM
python3-colorama	04.4	04.4
python3-configobj	50.6	5.0.6
python3-cryptography	36,0	36,0
python3-daemon	2.3.0	
python3-dateutil	28.1	28.1
python3-dbus	1,2,18	1,2,18
python3-distro	1.5.0	1.5.0
python3-dnf	4.14.0	4.14.0
python3-dnf-plugins-core	4.3.0	4.3.0
python3-docutils	0,16	0,16
python3-gpg	1.15.1	1.15.1
python3-hawkey	0,69,0	0,69,0
python3-idna	(2.10)	(2.10)
python3-jinja2	211,3	2.11.3
python3-jmespath	0.10.0	0.10.0
python3-jsonpatch	1,21	1,21
python3-jsonpointer	2.0	2.0
python3-jjsonschema	3.2.0	3.2.0
python3-libcomps	01,20	01,20
python3-libdnf	0,69,0	0,69,0

Pacchetto	AMI	KVM
python3-libs	3,9,16	3,9,16
python3-libselinux	3.4	3.4
python3-libsemanage	3.4	3.4
python3-libstorage mgmt	1.9.4	1.9.4
python3-lockfile	0.12.2	
python3-markupsafe	1.1.1	1.1.1
python3-netifaces	0,10,6	0,10,6
python3-oauthlib	3.0.2	3.0.2
python3-pip-wheel	21,31	21,31
python3-ply	3,11	3,11
python3-policycore utils	3.4	3.4
python3-prettytable	0.7.2	0.7.2
python3-prompt-too lkit	3,0,24	3,0,24
python3-pycparser	2,20	2,20
python3-pyrsistent	0,17,3	0,17,3
python3-pyserial	3.4	3.4
python3-pysocks	1.7.1	1.7.1
python3-pytz	2022,7,1	2022,7,1

Pacchetto	AMI	KVM
python3-pyyaml	5.4.1	5.4.1
python3-requests	2,25,1	2,25,1
python3-rpm	4,161,3	4,161,3
python3-ruamel-yaml	0,16,6	0,16,6
python3-ruamel-yaml-clib	0,12	0,1,2
python3-setools	4.4.1	4.4.1
python3-setuptools	59,6,0	59,6,0
python3-setuptools-wheel	59,6,0	59,6,0
python3-six	1.15.0	1.15.0
python3-systemd	235	235
python3-urllib3	1,25,10	1,25,10
python3-wcwidth	0,2,5	02,5
python-chevront	0.13.1	
python-srpm-macros	3.9	3.9
quota	4,06	4,06
quota-nls	4,06	4,06
readline	8.1	8.1
rng-tools	6,14	6,14
rootfiles	8.1	8.1

Pacchetto	AMI	KVM
rpcbind	1.2.6	1.2.6
rpm	4,16,13	4,161,3
rpm-build-libs	4,161,3	4,161,3
rpm-libs	4,161,3	4,161,3
rpm-plugin-selinux	4,161,3	4,161,3
rpm-plugin-systemd-inhibit	4,161,3	4,161,3
rpm-sign-libs	4,161,3	4,161,3
rsync	3.2.6	3.2.6
rust-srpm-macros	21	21
sbsigntools	0.9.4	0.9.4
screen	4.8.0	4.8.0
sed	4.8	4.8
selinux-policy	381,45	381,45
selinux-policy-targeted	381,45	381,45
setup	2,13,7	2,13,7
shadow-utils	4.9	4.9
slang	2.3.2	2.3.2
sqlite-libs	3,4,0	3,4,0
sssd-client	2,9,4	2,9,4

Pacchetto	AMI	KVM
sssd-common	2,9,4	
sssd-kcm	2,9,4	
sssd-nfs-idmap	2,9,4	
strace	6.8	6.8
sudo	1,9,15	1,9,15
sysctl-defaults	1.0	1.0
sysstat	12,5,6	12,5,6
systemd	252,23	252,23
systemd-libs	252,23	252,23
systemd-networkd	252,23	252,23
systemd-pam	252,23	252,23
systemd-resolved	252,23	252,23
systemd-udev	252,23	252,23
system-release	2023,620241031	2023,620241031
systemtap-runtime	4,8	4.8
tar	1,34	1,34
tbb	2020,3	2020,3
tcpdump	4,99,1	4,99,1
tcsh	6,24,07	6,24,07
time	1.9	1.9

Pacchetto	AMI	KVM
traceroute	2.1.3	2.1.3
tzdata	2024a	2024a
unzip	6.0	6.0
update-motd	2.2	2.2
userspace-rcu	0.12.1	0.12.1
util-linux	2,37,4	2,37,4
util-linux-core	2,37,4	2,37,4
vim-common	9,0,2153	9,0,2153
vim-data	9,0,2153	9,0,2153
vim-enhanced	9,0,2153	9,0,2153
vim-filesystem	9,0,2153	9,0,2153
vim-minimal	9,0,2153	9,0,2153
wget	1,21,3	1,21,3
which	2,21	2,21
words	3.0	3.0
xfsdump	3,1,11	3,1,11
xfspgrog	5,18,0	5,18,0
xxd	9,0,2153	9,0,2153
xxhash-libs	0.8.0	0.8.0
xz	52,5	5.2.5

Pacchetto	AMI	KVM
xz-libs	5.2.5	5.2.5
yum	4.14.0	4.14.0
zip	3.0	3.0
zlib	1,2,11	1,2,11
zram-generator	1.1.2	
zram-generator-def aults	1.1.2	
zstd	1,5,5	1,5,5

Confronto dei pacchetti installati sull'AMI standard di Amazon Linux 2023 con l' AL2immagine 023 VMware OVA

Un confronto tra il RPMs presente sull'AMI standard AL2 023 e il RPMs presente sull'immagine AL2 023 VMware OVA.

Pacchetto	AMI	VMware OVA
acl	2.3.1	2.3.1
acpid	2,0,32	
alternatives	1.15	1.15
amazon-chrony-config	4.3	
amazon-ec2-net-utils	2.5.1	
amazon-linux-onprem		1.2
amazon-linux-repo- cdn		2023,620241031

Pacchetto	AMI	VMware OVA
amazon-linux-repo-s3	2023,620241031	
amazon-linux-sb-keys	2023,1	2023,1
amazon-onprem-netw ork		1.2
amazon-rpm-config	228	228
amazon-ssm-agent	3,3,987,0	3,3,987,0
amd-ucode-firmware	20210208	20210208
at	3,1,23	3,1,23
attr	2.5.1	2.5.1
audit	30,6	3.0.6
audit-libs	3.0.6	3.0.6
aws-cfn-bootstrap	2.0	
awscli-2	2,15,30	2,15,30
basesystem	11	11
bash	5,2,15	5,2,15
bash-completion	2.11	2.11
bc	1,07,1	1,07,1
bind-libs	9,18,28	9,18,28
bind-license	9,18,28	9,18,28
bind-utils	9,18,28	9,18,28
binutils	2,39	2,39

Pacchetto	AMI	VMware OVA
boost-filesystem	1,75,0	1,75,0
boost-system	1,75,0	1,75,0
boost-thread	1,75,0	1,75,0
bzip2	1.0.8	1.0.8
bzip2-libs	1.0.8	1.0.8
ca-certificates	2023,2,68	2023,2,68
c-ares	1.19.1	
checkpolicy	3.4	3.4
chkconfig	1.15	1.15
chrony	4.3	4.3
cloud-init	222,2	222,2
cloud-init-cfg-ec2	222,2	
cloud-init-cfg-onpre		222,2
cloud-utils-growpart	0,31	0,31
coreutils	8,32	8,32
coreutils-common	8,32	8,32
cpio	2,13	2,13
cracklib	2,9,6	29.6
cracklib-dicts	29.6	29.6
crontabs	1.11	1.11

Pacchetto	AMI	VMware OVA
crypto-policies	20220428	20220428
crypto-policies-scripts	20220428	20220428
cryptsetup	2.6.1	2.6.1
cryptsetup-libs	2.6.1	2.6.1
curl-minimal	8,5,0	8,5,0
cyrus-sasl-lib	2,1,27	2,1,27
cyrus-sasl-plain	2,1,27	2,1,27
dbus	1,12,28	1,12,28
dbus-broker	32	32
dbus-common	1,12,28	1,12,28
dbus-libs	1,12,28	1,12,28
device-mapper	1,02,185	1,02,185
device-mapper-libs	1,02,185	1,02,185
diffutils	3.8	3.8
dnf	4.14.0	4.14.0
dnf-data	4.14.0	4.14.0
dnf-plugin-release-notification	1.2	1.2
dnf-plugins-core	4.3.0	4.3.0

Pacchetto	AMI	VMware OVA
dnf-plugin-support-info	1.2	1.2
dnf-utils	4.3.0	4.3.0
dosfstools	4.2	4.2
dracut	055	055
dracut-config-ec2	3.0	
dracut-config-generic	055	055
dwz	0,14	0,14
dyninst	102,1	10.2.1
e2fsprogs	1,446,5	1,46,5
e2fsprogs-libs	1,46,5	1,46,5
ec2-hibinit-agent	1.0.8	
ec2-instance-connect	1.1	
ec2-instance-connect-selinux	1.1	
ec2-utils	2.2.0	
ed	1.14.2	1.14.2
efi-filesystem	5	5
efi-srpm-macros	5	5
efivar	38	38

Pacchetto	AMI	VMware OVA
efivar-libs	38	38
elfutils-debuginfod-client	0.188	0.188
elfutils-default-yama-scope	0.188	0.188
elfutils-libelf	0.188	0.188
elfutils-libs	0.188	0.188
ethtool	5,15	5,15
expat	2.5.0	2.5.0
file	5,39	5,39
file-libs	5,39	5,39
filesystem	3,14	3,14
findutils	4.8.0	4.8.0
fonts-srpm-macros	2.0.5	2.0.5
fstrm	0.6.1	0.6.1
fuse3		3,10,4
fuse3-libs		3,10,4
fuse-common		3,10,4
fuse-libs	2,9,9	29.9
gawk	5.1.0	5.1.0
gdbm-libs	1,19	1,19

Pacchetto	AMI	VMware OVA
gdisk	1.0.8	1.0.8
gettext	0,21	0,21
gettext-libs	0,21	0,21
ghc-srpm-macros	1.5.0	1.5.0
glib2	2,74,7	2,74,7
glibc	2,34	2,34
glibc-all-langpacks	2,34	2,34
glibc-common	2,34	2,34
glibc-gconv-extra	2,34	2,34
glibc-locale-source	2,34	2,34
gmp	62,1	6.2.1
gnupg2-minimal	2.3.7	2.3.7
gnutls	3.8.0	3.8.0
go-srpm-macros	3.2.0	3.2.0
gpgme	1.15.1	1.15.1
gpm-libs	1,20.7	1,20.7
grep	3.8	3.8
groff-base	1,22,4	1,22,4
grub2-common	2,06	2,06
grub2-efi-x64-ec2	2,06	2,06

Pacchetto	AMI	VMware OVA
grub2-pc		2,06
grub2-pc-modules	2,06	2,06
grub2-tools	2,06	2,06
grub2-tools-minimal	2,06	2,06
grubby	8,40	8,40
gssproxy	0.8.4	0.8.4
gzip	1.12	1.12
hostname	3,23	3,23
hunspell	1.7.0	1.7.0
hunspell-en	0,20140811,1	0,20140811,1
hunspell-en-GB	0,20140811,1	0,20140811,1
hunspell-en-US	0,20140811,1	0,20140811,1
hunspell-filesystem	1.7.0	1.7.0
hwdata	0,384	0,384
info	6.7	6.7
inih	49	49
initscripts	10,09	10,09
iproute	6.10.0	6.10.0
iputils	20210202	20210202
irqbalance	1.9.0	1.9.0

Pacchetto	AMI	VMware OVA
jansson	2.14	2.14
jemalloc	52,1	5.2.1
jitterentropy	3.4.1	3.4.1
jq	1.7.1	
json-c	0,14	0,14
kbd	2.4.0	2.4.0
kbd-misc	2.4.0	2.4.0
kernel	6,1112	6,1112
kernel-libbpf	6,1112	6,1112
kernel-livepatch-r epo-cdn		2023,620241031
kernel-livepatch-r epo-s3	2023,620241031	
kernel-modules-extra		6,1112
kernel-modules-ext ra-common		6,1112
kernel-srpm-macros	1.0	1.0
kernel-tools	6,1112	6,1112
keyutils	1.6.3	1.6.3
keyutils-libs	1.6.3	1.6.3
kmod	29	29

Pacchetto	AMI	VMware OVA
kmod-libs	29	29
kpatch-runtime	0,9,7	0,9,7
krb5-libs	1,21,3	1,21,3
less	608	608
libacl	2.3.1	2.3.1
libaio	0,3111	0,3111
libarchive	3,7,4	3,7,4
libargon2	20171227	20171227
libassuan	2,5,5	2,5,5
libattr	2.5.1	2.5.1
libbasicobjects	0,11	01.1
libblkid	2,37,4	2,37,4
libcap	2,48	2,48
libcap-ng	0.8.2	0.8.2
libcbor	0.7.0	0.7.0
libcollection	0.7.0	0.7.0
libcom_err	1,446,5	1,46,5
libcomps	01,20	01,20
libconfig	17.2	1.7.2
libcurl-minimal	8,5,0	8,5,0

Pacchetto	AMI	VMware OVA
libdb	5,3,28	5,3,28
libdhash	0,50	
libdnf	0,69,0	0,69,0
libeconf	0,40	0,40
libedit	3.1	3.1
libev	4,33	4,33
libevent	2,1,12	2,1,12
libfdisk	2,37,4	2,37,4
libffi	34.4	34.4
libfido2	1.10.0	1.10.0
libgcc	11.4.1	11.4.1
libgcrypt	1.10.2	1.10.2
libgomp	11.4.1	11.4.1
libgpg-error	1,42	1,42
libibverbs	48,0	48,0
libidn2	2.3.2	2.3.2
libini_config	1.3.1	1.3.1
libkcapi	1.4.0	1.4.0
libkcapi-hmacalc	1.4.0	1.4.0
libldb	26.2	

Pacchetto	AMI	VMware OVA
libmaxminddb	1.5.2	1.5.2
libmetalink	0,13	0,13
libmnl	1.0.4	1.0.4
libmodulemd	2.13.0	2.13.0
libmount	2,37,4	2,37,4
libmspack		0,10,1
libnfsidmap	2,5,4	2.5.4
libnghttp2	1,59,0	1,59,0
libnl3	3.5.0	3.5.0
libpath_utils	0,2,1	0,2,1
libpcap	1.10.1	1.10.1
libpipeline	1.5.3	1.5.3
libpkgconf	1.8.0	1.8.0
libpsl	0,21,1	0,21,1
libpwquality	1.4.4	1.4.4
libref_array	0,1,5	0,1,5
librepo	1,14,5	1,14,5
libreport-filesystem	2,15,2	2,15,2
libseccomp	2.5.3	2.5.3
libselinux	3.4	3.4

Pacchetto	AMI	VMware OVA
libselinux-utils	3.4	3.4
libsemanage	3.4	3.4
libsepol	3.4	3.4
libsigsegv	2,13	2,13
libsmartcols	2,37,4	2,37,4
libsolv	0,7,22	0,7,22
libss	1,446,5	1,46,5
libsss_certmap	2,9,4	
libsss_idmap	2,9,4	2,9,4
libsss_nss_idmap	2,9,4	2,9,4
libsss_sudo	2,9,4	
libstdc++	11.4.1	11.4.1
libstoragegmt	1.9.4	1.9.4
libtalloc	2.3.4	
libtasn1	4,19,0	4,19,0
libtdb	1.4.7	
libtevent	0.13.0	
libtextstyle	0,21	0,21
libtirpc	1.3.3	1.3.3
libtool-ltdl		2.4.7

Pacchetto	AMI	VMware OVA
libunistring	0,9,10	0,9,10
libuser	0,63	0,63
libutempter	1.2.1	1.2.1
libuuid	2,37,4	2,37,4
libuv	1.47.0	1.47.0
libverto	0,32	0,32
libverto-libev	0,32	0,32
libxcrypt	4,4,33	4,4,33
libxml2	2,10,4	210.4
libxslt		1,1,34
libyaml	0,2,5	02,5
libzstd	1,5,5	1,5,5
linux-firmware-whe nce	20210208	20210208
lm_sensors-libs	3.6.0	3.6.0
lmdb-libs	0,9,29	0,9,29
logrotate	3,20,1	3,20,1
lsof	4,94,0	4,94,0
lua-libs	5.4.4	5.4.4
lua-srpm-macros	1	1
lz4-libs	1.9.4	1.9.4

Pacchetto	AMI	VMware OVA
man-db	2,9,3	29.3
man-pages	5,10	5,10
microcode_ctl	2.1	2.1
mpfr	4.1.0	4.1.0
nano	5.8	5.8
ncurses	6.2	6.2
ncurses-base	6.2	6.2
ncurses-libs	6.2	6.2
nettle	3.8	3.8
net-tools	2.0	2.0
newt	0,52,21	0,52,21
nfs-utils	2,5,4	2.5.4
npth	1.6	1.6
nspr	4,35,0	4,35,0
nss	3,90,0	3,90,0
nss-softokn	3,90,0	3,90,0
nss-softokn-freebl	3,90,0	3,90,0
nss-sysinit	3,90,0	3,90,0
nss-util	3,90,0	3,90,0
ntsysv	1.15	1.15

Pacchetto	AMI	VMware OVA
numactl-libs	2.0,14	2.0,14
ocaml-srpm-macros	6	6
oniguruma	6,9,7,1	
openblas-srpm-macros	2	2
openldap	2,4,57	2,4,57
openssh	8.7p1	8,7p1
openssh-clients	8,7p1	8,7p1
openssh-server	8,7p1	8,7p1
openssl	3,0,8	30,8
openssl-libs	30,8	30,8
openssl-pkcs11	0,4,12	0,4,12
open-vm-tools		12,3,0
os-prober	1,77	1,77
p11-kit	0,24,1	0,24,1
p11-kit-trust	0,24,1	0,24,1
package-notes-srpm-macros	0.4	0.4
pam	1.5.1	1.5.1
parted	3.4	3.4
passwd	0,80	0,80
pciutils	3.7.0	3.7.0

Pacchetto	AMI	VMware OVA
pciutils-libs	3.7.0	3.7.0
pcre2	10,40	10,40
pcre2-syntax	10,40	10,40
perl-Carp	1,50	1,50
perl-Class-Struct	0,66	0,66
perl-constant	1,33	1,33
perl-DynaLoader	1,47	1,47
perl-Encode	3,15	3,15
perl-Errno	1,30	1,30
perl-Exporter	5,74	5,74
perl-Fcntl	1.13	1.13
perl-File-Basename	2,85	2,85
perl-File-Path	2,18	2,18
perl-File-stat	1,09	1,09
perl-File-Temp	0,231,100	0,231,100
perl-Getopt-Long	2,52	2,52
perl-Getopt-Std	1.12	1.12
perl-HTTP-Tiny	0,078	0,078
perl-if	0,60,800	0,60,800
perl-interpreter	5,32,1	5,32,1

Pacchetto	AMI	VMware OVA
perl-I0	1,43	1,43
perl-IPC-Open3	1,21	1,21
perl-libs	5,32,1	5,32,1
perl-MIME-Base64	3,16	3,16
perl-mro	1,23	1,23
perl-overload	1,31	1,31
perl-overloading	0,02	0,02
perl-parent	0,238	0,238
perl-PathTools	3,78	3,78
perl-Pod-Escapes	1,07	1,07
perl-podlators	4,14	4,14
perl-Pod-Perldoc	3,28,01	3,28,01
perl-Pod-Simple	3,42	3,42
perl-Pod-Usage	2,01	2,01
perl-POSIX	1,94	1,94
perl-Scalar-List-Utils	1,56	1,56
perl-SelectSaver	1.02	1.02
perl-Socket	2,032	2,032
perl-srpm-macros	1	1
perl-Storable	3,21	3,21

Pacchetto	AMI	VMware OVA
perl-subst	1,03	1,03
perl-Symbol	1,08	1,08
perl-Term-ANSIColor	5,01	5,01
perl-Term-Cap	1,17	1,17
perl-Text-ParseWords	3,30	3,30
perl-Text-Tabs+Wrap	2021,0726	2021,0726
perl-Time-Local	1,300	1.300
perl-vars	1,05	1,05
pkgconf	1.8.0	1.8.0
pkgconf-m4	1.8.0	1.8.0
pkgconf-pkg-config	1.8.0	1.8.0
policycoreutils	3.4	3.4
policycoreutils-python-utils	3.4	
popt	1,18	1,18
procps-ng	3,3,17	3,3,17
protobuf-c	1.4.1	1.4.1
psacct	6,6,4	6.6.4
psmisc	23,4	23,4
publicsuffix-list-dafsa	20240212	20240212

Pacchetto	AMI	VMware OVA
python3	3,9,16	3,9,16
python3-attrs	20,3,0	20,3,0
python3-audit	30.6	3.0.6
python3-awscli	0,19,19	0,19,19
python3-babel	2,9,1	29.1
python3-cffi	1,14,5	1,14,5
python3-chardet	4.0.0	4.0.0
python3-colorama	04.4	04.4
python3-configobj	50.6	5.0.6
python3-cryptography	36,0	36,0
python3-daemon	2.3.0	
python3-dateutil	28.1	28.1
python3-dbus	1,2,18	1,2,18
python3-distro	1.5.0	1.5.0
python3-dnf	4.14.0	4.14.0
python3-dnf-plugins-core	4.3.0	4.3.0
python3-docutils	0,16	0,16
python3-gpg	1.15.1	1.15.1
python3-hawkey	0,69,0	0,69,0
python3-idna	(2.10)	(2.10)

Pacchetto	AMI	VMware OVA
python3-jinja2	211,3	2.11.3
python3-jmespath	0.10.0	0.10.0
python3-jsonpatch	1,21	1,21
python3-jsonpointer	2.0	2.0
python3-jsonschema	3.2.0	3.2.0
python3-libcomps	01,20	01,20
python3-libdnf	0,69,0	0,69,0
python3-libs	3,9,16	3,9,16
python3-libselenium	3.4	3.4
python3-libsemanage	3.4	3.4
python3-libstorage mgmt	1.9.4	1.9.4
python3-lockfile	0.12.2	
python3-markupsafe	1.1.1	1.1.1
python3-netifaces	0,10,6	0,10,6
python3-oauthlib	3.0.2	3.0.2
python3-pip-wheel	21,31	21,31
python3-ply	3,11	3,11
python3-policycore utils	3.4	3.4
python3-prettytable	0.7.2	0.7.2

Pacchetto	AMI	VMware OVA
python3-prompt-toolkit	3,0,24	3,0,24
python3-pycparser	2,20	2,20
python3-pyrsistent	0,17,3	0,17,3
python3-pyserial	3.4	3.4
python3-pysocks	1.7.1	1.7.1
python3-pytz	2022,7,1	2022,7,1
python3-pyyaml	5.4.1	5.4.1
python3-requests	2,25,1	2,25,1
python3-rpm	4,161,3	4,161,3
python3-ruamel-yaml	0,16,6	0,16,6
python3-ruamel-yaml- clib	0,12	0,1,2
python3-setools	4.4.1	4.4.1
python3-setuptools	59,60	59,6,0
python3-setuptools- wheel	59,6,0	59,6,0
python3-six	1.15.0	1.15.0
python3-systemd	235	235
python3-urllib3	1,25,10	1,25,10
python3-wcwidth	0,2,5	02,5

Pacchetto	AMI	VMware OVA
python-chevron	0.13.1	
python-srpm-macros	3.9	3.9
quota	4,06	4,06
quota-nls	4,06	4,06
readline	8.1	8.1
rng-tools	6,14	6,14
rootfiles	8.1	8.1
rpcbind	1.2.6	1.2.6
rpm	4,161,3	4,161,3
rpm-build-libs	4,161,3	4,161,3
rpm-libs	4,161,3	4,161,3
rpm-plugin-selinux	4,161,3	4,161,3
rpm-plugin-systemd-inhibit	4,161,3	4,161,3
rpm-sign-libs	4,161,3	4,161,3
rsync	3.2.6	3.2.6
rust-srpm-macros	21	21
sbsigntools	0.9.4	0.9.4
screen	4.8.0	4.8.0
sed	4.8	4.8
selinux-policy	381,45	381,45

Pacchetto	AMI	VMware OVA
selinux-policy-targeted	381,45	381,45
setup	2,13,7	2,13,7
shadow-utils	4.9	4.9
slang	2.3.2	2.3.2
sqlite-libs	3,4,0	3,4,0
sssd-client	2,9,4	2,9,4
sssd-common	2,9,4	
sssd-kcm	2,9,4	
sssd-nfs-idmap	2,9,4	
strace	6.8	6.8
sudo	1,9,15	1,9,15
sysctl-defaults	1.0	1.0
sysstat	12,5,6	12,5,6
systemd	252,23	252,23
systemd-libs	252,23	252,23
systemd-networkd	252,23	252,23
systemd-pam	252,23	252,23
systemd-resolved	252,23	252,23
systemd-udev	252,23	252,23
system-release	2023,620241031	2023,620241031

Pacchetto	AMI	VMware OVA
systemtap-runtime	4,8	4.8
tar	1,34	1,34
tbb	2020,3	2020,3
tcpdump	4,99,1	4,99,1
tcsh	6,24,07	6,24,07
time	1.9	1.9
traceroute	2.1.3	2.1.3
tzdata	2024a	2024a
unzip	6.0	6.0
update-motd	2.2	2.2
userspace-rcu	0.12.1	0.12.1
util-linux	2,37,4	2,37,4
util-linux-core	2,37,4	2,37,4
vim-common	9,0,2153	9,0,2153
vim-data	9,0,2153	9,0,2153
vim-enhanced	9,0,2153	9,0,2153
vim-filesystem	9,0,2153	9,0,2153
vim-minimal	9,0,2153	9,0,2153
wget	1,21,3	1,21,3
which	2,21	2,21

Pacchetto	AMI	VMware OVA
words	3.0	3.0
xfsdump	3,1,11	3,1,11
xfspgrog	5,18,0	5,18,0
xmlsec1		1,2,33
xmlsec1-openssl		1,2,33
xxd	9,0,2153	9,0,2153
xxhash-libs	0.8.0	0.8.0
xz	52,5	5.2.5
xz-libs	5.2.5	5.2.5
yum	4.14.0	4.14.0
zip	3.0	3.0
zlib	1,2,11	1,2,11
zram-generator	1.1.2	
zram-generator-def aults	1.1.2	
zstd	1,5,5	1,5,5

Confronto dei pacchetti installati sull'AMI standard di Amazon Linux 2023 con l'immagine AL2 Hyper-V 023

Un confronto tra la versione RPMs attuale dell'AMI standard AL2 023 e quella RPMs presente sull'immagine Hyper-V AL2 023.

Pacchetto	AMI	Hyper-V VHDX
acl	2.3.1	2.3.1
acpid	2,0,32	
alternatives	1.15	1.15
amazon-chroney-config	4.3	
amazon-ec2-net-utils	2.4.1	
amazon-linux-onprem		1.2
amazon-linux-repo-cdn		2023,420240319
amazon-linux-repo-s3	2023,420240319	
amazon-linux-sb-keys	2023,1	2023,1
amazon-onprem-network		1.2
amazon-rpm-config	228	228
amazon-ssm-agent	3,2,2303,0	3,2233,0
at	3,1,23	3,1,23
attr	2.5.1	2.5.1
audit	30,6	3.0.6
audit-libs	3.0.6	3.0.6
aws-cfn-bootstrap	2.0	
awscli-2	2.14.5	2.14.5
basesystem	11	11

Pacchetto	AMI	Hyper-V VHDX
bash	5,2,15	5,2,15
bash-completion	2.11	2.11
bc	1,07,1	1,07,1
bind-libs	9,16,48	9,16,48
bind-license	9,16,48	9,16,48
bind-utils	9,16,48	9,16,48
binutils	2,39	2,39
boost-filesystem	1,75,0	1,75,0
boost-system	1,75,0	1,75,0
boost-thread	1,75,0	1,75,0
bzip2	1.0.8	1.0.8
bzip2-libs	1.0.8	1.0.8
ca-certificates	2023,2,64	2023,2,64
c-ares	1.19.0	
checkpolicy	3.4	3.4
chkconfig	1.15	1.15
chrony	4.3	4.3
cloud-init	222,2	222,2
cloud-init-cfg-ec2	222,2	
cloud-init-cfg-onpre		222,2

Pacchetto	AMI	Hyper-V VHDX
cloud-utils-growpart	0,31	0,31
coreutils	8,32	8,32
coreutils-common	8,32	8,32
cpio	2,13	2,13
cracklib	2,9,6	29.6
cracklib-dicts	29.6	29.6
crontabs	1.11	1.11
crypto-policies	20220428	20220428
crypto-policies-scripts	20220428	20220428
cryptsetup	2.6.1	2.6.1
cryptsetup-libs	2.6.1	2.6.1
curl-minimal	8,5,0	8,5,0
cyrus-sasl-lib	2,1,27	2,1,27
cyrus-sasl-plain	2,1,27	2,1,27
dbus	1,12,28	1,12,28
dbus-broker	32	32
dbus-common	1,12,28	1,12,28
dbus-libs	1,12,28	1,12,28
device-mapper	1,02,185	1,02,185
device-mapper-libs	1,02,185	1,02,185

Pacchetto	AMI	Hyper-V VHDX
diffutils	3.8	3.8
dnf	4.14.0	4.14.0
dnf-data	4.14.0	4.14.0
dnf-plugin-release-notification	1.2	1.2
dnf-plugins-core	4.3.0	4.3.0
dnf-plugin-support-info	1.2	1.2
dnf-utils	4.3.0	4.3.0
dosfstools	4.2	4.2
dracut	055	055
dracut-config-ec2	3.0	
dracut-config-generic	055	055
dwz	0,14	0,14
dyninst	102,1	10.2.1
e2fsprogs	1,446,5	1,46,5
e2fsprogs-libs	1,46,5	1,46,5
ec2-hibinit-agent	1.0.8	
ec2-instance-connect	1.1	
ec2-instance-connect-selinux	1.1	

Pacchetto	AMI	Hyper-V VHDX
ec2-utils	2.2.0	
ed	1.14.2	1.14.2
efi-filesystem	5	5
efi-srpm-macros	5	5
efivar	38	38
efivar-libs	38	38
elfutils-debuginfod-client	0.188	0.188
elfutils-default-yama-scope	0.188	0.188
elfutils-libelf	0.188	0.188
elfutils-libs	0.188	0.188
ethtool	5,15	5,15
expat	2.5.0	2.5.0
file	5,39	5,39
file-libs	5,39	5,39
filesystem	3,14	3,14
findutils	4.8.0	4.8.0
fonts-srpm-macros	2.0.5	2.0.5
fstrm	0.6.1	0.6.1
fuse-libs	2,9,9	29.9

Pacchetto	AMI	Hyper-V VHDX
gawk	5.1.0	5.1.0
gdbm-libs	1,19	1,19
gdisk	1.0.8	1.0.8
gettext	0,21	0,21
gettext-libs	0,21	0,21
ghc-srpm-macros	1.5.0	1.5.0
glib2	2,74,7	2,74,7
glibc	2,34	2,34
glibc-all-langpacks	2,34	2,34
glibc-common	2,34	2,34
glibc-gconv-extra	2,34	2,34
glibc-locale-source	2,34	2,34
gmp	62,1	6.2.1
gnupg2-minimal	2.3.7	2.3.7
gnutls	3.8.0	3.8.0
go-srpm-macros	3.2.0	3.2.0
gpgme	1.15.1	1.15.1
gpm-libs	1,20.7	1,20.7
grep	3.8	3.8
groff-base	1,22,4	1,22,4

Pacchetto	AMI	Hyper-V VHDX
grub2-common	2,06	2,06
grub2-efi-x64-ec2	2,06	2,06
grub2-pc		2,06
grub2-pc-modules	2,06	2,06
grub2-tools	2,06	2,06
grub2-tools-minimal	2,06	2,06
grubby	8,40	8,40
gssproxy	0.8.4	0.8.4
gzip	1.12	1.12
hostname	3,23	3,23
hunspell	1.7.0	1.7.0
hunspell-en	0,20140811,1	0,20140811,1
hunspell-en-GB	0,20140811,1	0,20140811,1
hunspell-en-US	0,20140811,1	0,20140811,1
hunspell-filesystem	1.7.0	1.7.0
hwdata	0,3353	0,3353
hyperv-daemons		0
hyperv-daemons-lic ense		0
hypervfcopyd		0
hypervkvpd		0

Pacchetto	AMI	Hyper-V VHDX
hyperv-tools		0
hypervvssd		0
info	6.7	6.7
inih	49	49
initscripts	10,09	10,09
iproute	5.10.0	5.10.0
iputils	20210202	20210202
irqbalance	1.9.0	1.9.0
jansson	2.14	2.14
jitterentropy	3.4.1	3.4.1
jq	1.7.1	1.7.1
json-c	0,14	0,14
kbd	2.4.0	2.4.0
kbd-misc	2.4.0	2.4.0
kernel	61,79	6,1,79
kernel-livepatch-r epo-cdn		2023,420240319
kernel-livepatch-r epo-s3	2023,420240319	
kernel-modules-extra		6,1,79

Pacchetto	AMI	Hyper-V VHDX
kernel-modules-extra-common		6,1,79
kernel-srpm-macros	1.0	1.0
kernel-tools	6,1,79	6,1,79
keyutils	1.6.3	1.6.3
keyutils-libs	1.6.3	1.6.3
kmod	29	29
kmod-libs	29	29
kpatch-runtime	0,9,7	0,9,7
krb5-libs	1,2,1	1,2,1
less	608	608
libacl	2.3.1	2.3.1
libaio	0,3111	0,3111
libarchive	3,5,3	3,5,3
libargon2	20171227	20171227
libassuan	2,5,5	2,5,5
libattr	2.5.1	2.5.1
libbasicobjects	0,11	01.1
libblkid	2,37,4	2,37,4
libcap	2,48	2,48
libcap-ng	0.8.2	0.8.2

Pacchetto	AMI	Hyper-V VHDX
libcbor	0.7.0	0.7.0
libcollection	0.7.0	0.7.0
libcom_err	1,446,5	1,46,5
libcomps	01,20	01,20
libconfig	17.2	1.7.2
libcurl-minimal	8,5,0	8,5,0
libdb	5,3,28	5,3,28
libdhash	0,50	
libdnf	0,69,0	0,69,0
libeconf	0,40	0,40
libedit	3.1	3.1
libev	4,33	4,33
libevent	2,1,12	2,1,12
libfdisk	2,37,4	2,37,4
libffi	34.4	34.4
libfido2	1.10.0	1.10.0
libgcc	114,1	11.4.1
libgcrypt	1.10.2	1.10.2
libgomp	11.4.1	11.4.1
libgpg-error	1,42	1,42

Pacchetto	AMI	Hyper-V VHDX
libibverbs	48,0	48,0
libidn2	2.3.2	2.3.2
libini_config	1.3.1	1.3.1
libkcap	1.4.0	1.4.0
libkcap-hmacalc	1.4.0	1.4.0
libldb	26.2	
libmaxminddb	1.5.2	1.5.2
libmetalink	0,13	0,13
libmnl	1.0.4	1.0.4
libmodulemd	2.13.0	2.13.0
libmount	2,37,4	2,37,4
libnfsidmap	2,5,4	2.5.4
libnghttp2	1,57,0	1,57,0
libnl3	3.5.0	3.5.0
libpath_utils	0,2,1	0,2,1
libpcap	1.10.1	1.10.1
libpipeline	1.5.3	1.5.3
libpkgconf	1.8.0	1.8.0
libpsl	0,21,1	0,21,1
libpwquality	1.4.4	1.4.4

Pacchetto	AMI	Hyper-V VHDX
libref_array	0,1,5	0,1,5
librepo	1,14,5	1,14,5
libreport-filessystem	2,15,2	2,15,2
libseccomp	2.5.3	2.5.3
libselinux	3.4	3.4
libselinux-utils	3.4	3.4
libsemanage	3.4	3.4
libsepol	3.4	3.4
libsigsegv	2,13	2,13
libsmartcols	2,37,4	2,37,4
libsolv	0,7,22	0,7,22
libss	1,446,5	1,46,5
libsss_certmap	2,9,4	
libsss_idmap	2,9,4	2,9,4
libsss_nss_idmap	2,9,4	2,9,4
libsss_sudo	2,9,4	
libstdc++	11.4.1	11.4.1
libstoragegmt	1.9.4	1.9.4
libtalloc	2.3.4	
libtasn1	4,19,0	4,19,0

Pacchetto	AMI	Hyper-V VHDX
libtdb	1.4.7	
libtevent	0.13.0	
libtextstyle	0,21	0,21
libtirpc	1.3.3	1.3.3
libunistring	0,9,10	0,9,10
libuser	0,63	0,63
libutempter	1.2.1	1.2.1
libuuid	2,37,4	2,37,4
libuv	1.47.0	1.47.0
libverto	0,32	0,32
libverto-libev	0,32	0,32
libxcrypt	4,4,33	4,4,33
libxml2	2,10,4	210.4
libyaml	02,5	02,5
libzstd	1,5,5	1,5,5
lm_sensors-libs	3.6.0	3.6.0
lmdb-libs	0,9,29	0,9,29
logrotate	3,20,1	3,20,1
lsof	4,94,0	4,94,0
lua-libs	54.4	5.4.4

Pacchetto	AMI	Hyper-V VHDX
lua-srpm-macros	1	1
lz4-libs	1.9.4	1.9.4
man-db	2,9,3	29.3
man-pages	5,10	5,10
microcode_ctl	2.1	2.1
mpfr	4.1.0	4.1.0
nano	5.8	5.8
ncurses	6.2	6.2
ncurses-base	6.2	6.2
ncurses-libs	6.2	6.2
nettle	3.8	3.8
net-tools	2.0	2.0
newt	0,52,21	0,52,21
nfs-utils	2,5,4	2.5.4
npth	1.6	1.6
nspr	4,35,0	4,35,0
nss	3,90,0	3,90,0
nss-softokn	3,90,0	3,90,0
nss-softokn-freebl	3,90,0	3,90,0
nss-sysinit	3,90,0	3,90,0

Pacchetto	AMI	Hyper-V VHDX
nss-util	3,90,0	3,90,0
ntsysv	1.15	1.15
numactl-libs	2.0,14	2.0,14
ocaml-srpm-macros	6	6
oniguruma	6,9,7,1	6,9,7,1
openblas-srpm-macros	2	2
openldap	2,4,57	2,4,57
openssh	8.7p1	8,7p1
openssh-clients	8,7p1	8,7p1
openssh-server	8,7p1	8,7p1
openssl	3,0,8	30,8
openssl-libs	30,8	30,8
openssl-pkcs11	0,4,12	0,4,12
os-prober	1,77	1,77
p11-kit	0,24,1	0,24,1
p11-kit-trust	0,24,1	0,24,1
package-notes-srpm-macros	0.4	0.4
pam	1.5.1	1.5.1
parted	3.4	3.4
passwd	0,80	0,80

Pacchetto	AMI	Hyper-V VHDX
pciutils	3.7.0	3.7.0
pciutils-libs	3.7.0	3.7.0
pcre2	10,40	10,40
pcre2-syntax	10,40	10,40
perl-Carp	1,50	1,50
perl-Class-Struct	0,66	0,66
perl-constant	1,33	1,33
perl-DynaLoader	1,47	1,47
perl-Encode	3,15	3,15
perl-Errno	1,30	1,30
perl-Exporter	5,74	5,74
perl-Fcntl	1.13	1.13
perl-File-Basename	2,85	2,85
perl-File-Path	2,18	2,18
perl-File-stat	1,09	1,09
perl-File-Temp	0,231,100	0,231,100
perl-Getopt-Long	2,52	2,52
perl-Getopt-Std	1.12	1.12
perl-HTTP-Tiny	0,078	0,078
perl-if	0,60,800	0,60,800

Pacchetto	AMI	Hyper-V VHDX
perl-interpreter	5,32,1	5,32,1
perl-IO	1,43	1,43
perl-IPC-Open3	1,21	1,21
perl-libs	5,32,1	5,32,1
perl-MIME-Base64	3,16	3,16
perl-mro	1,23	1,23
perl-overload	1,31	1,31
perl-overloading	0,02	0,02
perl-parent	0,238	0,238
perl-PathTools	3,78	3,78
perl-Pod-Escapes	1,07	1,07
perl-podlators	4,14	4,14
perl-Pod-Perldoc	3,28,01	3,28,01
perl-Pod-Simple	3,42	3,42
perl-Pod-Usage	2,01	2,01
perl-POSIX	1,94	1,94
perl-Scalar-List-Utils	1,56	1,56
perl-SelectSaver	1.02	1.02
perl-Socket	2,032	2,032
perl-srpm-macros	1	1

Pacchetto	AMI	Hyper-V VHDX
perl-Storable	3,21	3,21
perl-subst	1,03	1,03
perl-Symbol	1,08	1,08
perl-Term-ANSIColor	5,01	5,01
perl-Term-Cap	1,17	1,17
perl-Text-ParseWords	3,30	3,30
perl-Text-Tabs+Wrap	2021,0726	2021,0726
perl-Time-Local	1,300	1.300
perl-vars	1,05	1,05
pkgconf	1.8.0	1.8.0
pkgconf-m4	1.8.0	1.8.0
pkgconf-pkg-config	1.8.0	1.8.0
policycoreutils	3.4	3.4
policycoreutils-python-utils	3.4	
popt	1,18	1,18
procps-ng	3,3,17	3,3,17
protobuf-c	1.4.1	1.4.1
psacct	6,6,4	6.6.4
psmisc	23,4	23,4

Pacchetto	AMI	Hyper-V VHDX
publicsuffix-list-dafsa	20240212	20240212
python3	3,9,16	3,9,16
python3-attrs	20,3,0	20,3,0
python3-audit	30.6	3.0.6
python3-awscrt	0,19,19	0,19,19
python3-babel	2,9,1	29.1
python3-cffi	1,14,5	1,14,5
python3-chardet	4.0.0	4.0.0
python3-colorama	04.4	04.4
python3-configobj	50.6	5.0.6
python3-cryptography	36,0	36,0
python3-daemon	2.3.0	
python3-dateutil	28.1	28.1
python3-dbus	1,2,18	1,2,18
python3-distro	1.5.0	1.5.0
python3-dnf	4.14.0	4.14.0
python3-dnf-plugins-core	4.3.0	4.3.0
python3-docutils	0,16	0,16
python3-gpg	1.15.1	1.15.1

Pacchetto	AMI	Hyper-V VHDX
python3-hawkey	0,69,0	0,69,0
python3-idna	(2.10)	(2.10)
python3-jinja2	211,3	2.11.3
python3-jmespath	0.10.0	0.10.0
python3-jsonpatch	1,21	1,21
python3-jsonpointer	2.0	2.0
python3-jsonschema	3.2.0	3.2.0
python3-libcomps	01,20	01,20
python3-libdnf	0,69,0	0,69,0
python3-libs	3,9,16	3,9,16
python3-libselenium	3.4	3.4
python3-libsemanage	3.4	3.4
python3-libstorage mgmt	1.9.4	1.9.4
python3-lockfile	0.12.2	
python3-markupsafe	1.1.1	1.1.1
python3-netifaces	0,10,6	0,10,6
python3-oauthlib	3.0.2	3.0.2
python3-pip-wheel	21,31	21,31
python3-ply	3,11	3,11

Pacchetto	AMI	Hyper-V VHDX
python3-policycore utils	3.4	3.4
python3-prettytable	0.7.2	0.7.2
python3-prompt-too lkit	3,0,24	3,0,24
python3-pycparser	2,20	2,20
python3-pyrsistent	0,17,3	0,17,3
python3-pyserial	3.4	3.4
python3-pysocks	1.7.1	1.7.1
python3-pytz	2022,7,1	2022,7,1
python3-pyyaml	5.4.1	5.4.1
python3-requests	2,25,1	2,25,1
python3-rpm	4,161,3	4,161,3
python3-ruamel-yaml	0,16,6	0,16,6
python3-ruamel-yaml- clib	0,12	0,1,2
python3-setools	4.4.1	4.4.1
python3-setuptools	59,6,0	59,6,0
python3-setuptools- wheel	59,6,0	59,6,0
python3-six	1.15.0	1.15.0
python3-systemd	235	235

Pacchetto	AMI	Hyper-V VHDX
python3-urllib3	1,25,10	1,25,10
python3-wcwidth	0,2,5	02,5
python-chevron	0.13.1	
python-srpm-macros	3.9	3.9
quota	4,06	4,06
quota-nls	4,06	4,06
readline	8.1	8.1
rng-tools	6,14	6,14
rootfiles	8.1	8.1
rpcbind	1.2.6	1.2.6
rpm	4,16,13	4,161,3
rpm-build-libs	4,161,3	4,161,3
rpm-libs	4,161,3	4,161,3
rpm-plugin-selinux	4,161,3	4,161,3
rpm-plugin-systemd-inhibit	4,161,3	4,161,3
rpm-sign-libs	4,161,3	4,161,3
rsync	3.2.6	3.2.6
rust-srpm-macros	21	21
sbsigntools	0.9.4	0.9.4
screen	4.8.0	4.8.0

Pacchetto	AMI	Hyper-V VHDX
sed	4.8	4.8
selinux-policy	37,22	37,22
selinux-policy-targeted	37,22	37,22
setup	2,13,7	2,13,7
shadow-utils	4.9	4.9
slang	2.3.2	2.3.2
sqlite-libs	3,4,0	3,4,0
sssd-client	2,9,4	2,9,4
sssd-common	2,9,4	
sssd-kcm	2,9,4	
sssd-nfs-idmap	2,9,4	
strace	5,16	5,16
sudo	1,9,14	1,9,14
sysctl-defaults	1.0	1.0
sysstat	12,5,6	12,5,6
systemd	252,16	252,16
systemd-libs	252,16	252,16
systemd-networkd	252,16	252,16
systemd-pam	252,16	252,16
systemd-resolved	252,16	252,16

Pacchetto	AMI	Hyper-V VHDX
systemd-udev	252,16	252,16
system-release	2023,420240319	2023,420240319
systemtap-runtime	4,8	4.8
tar	1,34	1,34
tbb	2020,3	2020,3
tcpdump	4,99,1	4,99,1
tcsch	6,24,07	6,24,07
time	1.9	1.9
traceroute	2.1.3	2.1.3
tzdata	2024a	2024a
unzip	6.0	6.0
update-motd	2.2	2.2
userspace-rcu	0.12.1	0.12.1
util-linux	2,37,4	2,37,4
util-linux-core	2,37,4	2,37,4
vim-common	9,0,2153	9,0,2153
vim-data	9,0,2153	9,0,2153
vim-enhanced	9,0,2153	9,0,2153
vim-filesystem	9,0,2153	9,0,2153
vim-minimal	9,0,2153	9,0,2153

Pacchetto	AMI	Hyper-V VHDX
wget	1,21,3	1,21,3
which	2,21	2,21
words	3.0	3.0
xfsdump	3,1,11	3,1,11
xfspgrog	5,18,0	5,18,0
xxd	9,0,2153	9,0,2153
xxhash-libs	0.8.0	0.8.0
xz	52,5	5.2.5
xz-libs	5.2.5	5.2.5
yum	4.14.0	4.14.0
zip	3.0	3.0
zlib	1,2,11	1,2,11
zram-generator	1.1.2	
zram-generator-def aults	1.1.2	
zstd	1,5,5	1,5,5

Identificazione di istanze e versioni di Amazon Linux

Può essere importante essere in grado di determinare quale distribuzione Linux e quale versione di tale distribuzione è un'immagine o un'istanza del sistema operativo. Amazon Linux fornisce meccanismi per distinguere Amazon Linux dalle altre distribuzioni Linux, nonché per identificare a quale versione di Amazon Linux si riferisce l'immagine.

Questa sezione tratterà i diversi metodi che possono essere utilizzati, i relativi limiti e esaminerà alcuni esempi del loro utilizzo.

Argomenti

- [Utilizzo dello os-release standard](#)
- [Specifico per Amazon Linux](#)
- [Codice di esempio per il rilevamento del sistema operativo](#)

Utilizzo dello **os-release** standard

Amazon Linux è conforme allo [os-releasestandard](#) per l'identificazione delle distribuzioni Linux. Questo file fornisce informazioni leggibili da un computer sull'identificazione del sistema operativo e sulla versione.

Note

Lo standard impone che `/etc/os-release` si tenti di essere analizzato per primo, seguito da `/usr/lib/os-release`. È necessario prestare attenzione a seguire lo standard relativo ai nomi e ai percorsi dei file.

Argomenti

- [Principali differenze di identificazione](#)
- [Tipi di campo: leggibile dalla macchina vs. leggibile dall'uomo](#)
- [Esempi di `/etc/os-release`](#)
- [Confronto con altre distribuzioni](#)

Principali differenze di identificazione

`os-release` Si trova in `/etc/os-release`, e se non è presente, in `/usr/lib/os-release`. Consulta lo [os-releasestandard](#) per informazioni complete.

Il modo più affidabile per determinare se un'istanza sta eseguendo Amazon Linux è quello di inserire il ID campo `os-release`.

Il modo più affidabile per distinguere tra le versioni consiste nel controllare il `VERSION_ID` campo in `os-release`:

- AMI Amazon Linux: `VERSION_ID` contiene una versione basata sulla data (ad es.) `2018.03`
- AL2: `VERSION_ID="2"`
- AL2023: `VERSION_ID="2023"`

Note

Ricorda che `VERSION_ID` è un campo leggibile da una macchina destinato all'uso programmatico, mentre `PRETTY_NAME` è progettato per essere visualizzato dagli utenti. [the section called "Tipi di campo"](#) Per ulteriori informazioni sui tipi di campo, vedere.

Tipi di campo: leggibile dalla macchina vs. leggibile dall'uomo

Il `/etc/os-release` file (o `/usr/lib/os-release` se `/etc/os-release` non esiste) contiene due tipi di campi: campi leggibili da computer destinati all'uso programmatico e campi leggibili dall'uomo destinati alla presentazione agli utenti.

Campi leggibili dalla macchina

Questi campi utilizzano formati standardizzati e sono destinati all'elaborazione mediante script, gestori di pacchetti e altri strumenti automatizzati. Contengono solo lettere minuscole, numeri e punteggiatura limitata (punti, trattini bassi e trattini).

- ID— Identificatore del sistema operativo. Amazon Linux lo utilizza `amzn` in tutte le versioni, distinguendolo da altre distribuzioni come Debian (`debian`), Ubuntu (`ubuntu`) o Fedora (`fedora`)

- `VERSION_ID`— Versione del sistema operativo per uso programmatico (ad es.) `2023`
- `ID_LIKE`— Elenco separato da spazi delle distribuzioni correlate (ad es.) `fedora`
- `VERSION_CODENAME`— Nome in codice di rilascio per gli script (ad es.) `karoo`
- `VARIANT_ID`— Identificatore di variante per le decisioni programmatiche
- `BUILD_ID`— Crea un identificatore per le immagini di sistema
- `IMAGE_ID`— Identificatore di immagine per ambienti containerizzati
- `PLATFORM_ID`— Identificatore della piattaforma (ad es.) `platform:al2023`

Campi leggibili dall'uomo

Questi campi sono destinati alla visualizzazione da parte degli utenti e possono contenere spazi, lettere maiuscole e minuscole e testo descrittivo. Dovrebbero essere usati quando si presentano informazioni sul sistema operativo nelle interfacce utente.

- `NAME`— Nome del sistema operativo da visualizzare (ad esempio, `Amazon Linux`)
- `PRETTY_NAME`— Nome completo del sistema operativo con versione da visualizzare (ad es. `Amazon Linux 2023.8.20250721`)
- `VERSION`— Informazioni sulla versione adatte alla presentazione all'utente
- `VARIANT`— Nome della variante o dell'edizione da visualizzare (ad es. `Server Edition`)

Altri campi informativi

Questi campi forniscono metadati aggiuntivi sul sistema operativo:

- `HOME_URL`— URL della home page del progetto
- `DOCUMENTATION_URL`— URL della documentazione
- `SUPPORT_URL`— URL delle informazioni di supporto
- `BUG_REPORT_URL`— URL di segnalazione dei bug
- `VENDOR_NAME`— Nome del fornitore
- `VENDOR_URL`— URL del fornitore
- `SUPPORT_END`— End-of-support data in formato `YYYY-MM-DD`
- `CPE_NAME`— Identificatore comune di enumerazione della piattaforma
- `ANSI_COLOR`— Codice a colori ANSI per la visualizzazione del terminale

Quando scrivi script o applicazioni che devono identificare Amazon Linux a livello di codice, usa campi leggibili dalla macchina come `e. ID VERSION_ID`. Quando mostri informazioni sul sistema operativo agli utenti, usa campi leggibili dall'uomo come `PRETTY_NAME`.

Esempi di `/etc/os-release`

Il contenuto dei `/etc/os-release` file varia tra le versioni di Amazon Linux:

AL2023

```
[ec2-user ~]$ cat /etc/os-release
```

```
NAME="Amazon Linux"
VERSION="2023"
ID="amzn"
ID_LIKE="fedora"
VERSION_ID="2023"
PLATFORM_ID="platform:al2023"
PRETTY_NAME="Amazon Linux 2023.8.20250721"
ANSI_COLOR="0;33"
CPE_NAME="cpe:2.3:o:amazon:amazon_linux:2023"
HOME_URL="https://aws.amazon.com/linux/amazon-linux-2023/"
DOCUMENTATION_URL="https://docs.aws.amazon.com/linux/"
SUPPORT_URL="https://aws.amazon.com/premiumsupport/"
BUG_REPORT_URL="https://github.com/amazonlinux/amazon-linux-2023"
VENDOR_NAME="AWS"
VENDOR_URL="https://aws.amazon.com/"
SUPPORT_END="2029-06-30"
```

AL2

```
[ec2-user ~]$ cat /etc/os-release
```

```
NAME="Amazon Linux"
VERSION="2"
ID="amzn"
ID_LIKE="centos rhel fedora"
VERSION_ID="2"
PRETTY_NAME="Amazon Linux 2"
ANSI_COLOR="0;33"
```

```
CPE_NAME="cpe:2.3:o:amazon:amazon_linux:2"  
HOME_URL="https://amazonlinux.com/"  
SUPPORT_END="2026-06-30"
```

Amazon Linux AMI

```
[ec2-user ~]$ cat /etc/os-release
```

```
NAME="Amazon Linux AMI"  
VERSION="2018.03"  
ID="amzn"  
ID_LIKE="rhel fedora"  
VERSION_ID="2018.03"  
PRETTY_NAME="Amazon Linux AMI 2018.03"  
ANSI_COLOR="0;33"  
CPE_NAME="cpe:/o:amazon:linux:2018.03:ga"  
HOME_URL="http://aws.amazon.com/amazon-linux-ami/"
```

Confronto con altre distribuzioni

Per capire come Amazon Linux si inserisce nel più ampio ecosistema Linux, confronta il suo `/etc/os-release` formato con le altre principali distribuzioni:

Fedora

```
[ec2-user ~]$ cat /etc/os-release
```

```
NAME="Fedora Linux"  
VERSION="42 (Container Image)"  
RELEASE_TYPE=stable  
ID=fedora  
VERSION_ID=42  
VERSION_CODENAME=""  
PLATFORM_ID="platform:f42"  
PRETTY_NAME="Fedora Linux 42 (Container Image)"  
ANSI_COLOR="0;38;2;60;110;180"  
LOGO=fedora-logo-icon  
CPE_NAME="cpe:/o:fedoraproject:fedora:42"  
DEFAULT_HOSTNAME="fedora"  
HOME_URL="https://fedoraproject.org/"
```

```
DOCUMENTATION_URL="https://docs.fedoraproject.org/en-US/fedora/f42/system-
administrators-guide/"
SUPPORT_URL="https://ask.fedoraproject.org/"
BUG_REPORT_URL="https://bugzilla.redhat.com/"
REDHAT_BUGZILLA_PRODUCT="Fedora"
REDHAT_BUGZILLA_PRODUCT_VERSION=42
REDHAT_SUPPORT_PRODUCT="Fedora"
REDHAT_SUPPORT_PRODUCT_VERSION=42
SUPPORT_END=2026-05-13
VARIANT="Container Image"
VARIANT_ID=container
```

Debian

```
[ec2-user ~]$ cat /etc/os-release
```

```
PRETTY_NAME="Debian GNU/Linux 12 (bookworm)"
NAME="Debian GNU/Linux"
VERSION_ID="12"
VERSION="12 (bookworm)"
VERSION_CODENAME=bookworm
ID=debian
HOME_URL="https://www.debian.org/"
SUPPORT_URL="https://www.debian.org/support"
BUG_REPORT_URL="https://bugs.debian.org/"
```

Ubuntu

```
[ec2-user ~]$ cat /etc/os-release
```

```
PRETTY_NAME="Ubuntu 24.04.2 LTS"
NAME="Ubuntu"
VERSION_ID="24.04"
VERSION="24.04.2 LTS (Noble Numbat)"
VERSION_CODENAME=noble
ID=ubuntu
ID_LIKE=debian
HOME_URL="https://www.ubuntu.com/"
SUPPORT_URL="https://help.ubuntu.com/"
BUG_REPORT_URL="https://bugs.launchpad.net/ubuntu/"
PRIVACY_POLICY_URL="https://www.ubuntu.com/legal/terms-and-policies/privacy-policy"
UBUNTU_CODENAME=noble
```

```
LOGO=ubuntu-logo
```

Nota come i campi leggibili dalla macchina forniscono un'identificazione coerente tra le distribuzioni:

- **ID**— Identifica in modo univoco il sistema operativo: per amzn Amazon Linux, fedora per Fedora, per Debian, debian per Ubuntu ubuntu
- **ID_LIKE**— Mostra le relazioni di distribuzione: Amazon Linux usa fedora (AL2023) o centos rhel fedora (AL2), mentre Ubuntu mostra debian per indicare la sua eredità Debian
- **VERSION_ID**— Fornisce informazioni sulla versione analizzabili automaticamente: 2023 per AL2 023, per Fedora, per Debian, 42 per Ubuntu 12 24.04

Al contrario, i campi leggibili dall'uomo sono progettati per essere visualizzati agli utenti:

- **NAME**— Nome del sistema operativo intuitivo:,, Amazon Linux Fedora Linux Debian GNU/Linux Ubuntu
- **PRETTY_NAME**— Nome visualizzato completo con versione:Amazon Linux 2023.8.20250721,,Fedora Linux 42 (Container Image), Debian GNU/Linux 12 (bookworm) Ubuntu 24.04.2 LTS
- **VERSION**— Versione leggibile dall'uomo con contesto aggiuntivo come nomi in codice o tipi di versione

Quando scrivi script multiplatforma, usa sempre i campi leggibili dalla macchina (, IDVERSION_ID,ID_LIKE) per la logica e le decisioni e usa i campi leggibili dall'uomo (,) solo per mostrare informazioni agli utenti. PRETTY_NAME NAME

Specifico per Amazon Linux

Esistono alcuni file specifici di Amazon Linux che possono essere utilizzati per identificare Amazon Linux e la sua versione. Il nuovo codice deve utilizzare lo [/etc/os-release](#) standard per essere compatibile con più distribuzioni. L'uso di file specifici di Amazon Linux è sconsigliato.

Argomenti

- [Il file /etc/system-release](#)
- [File di identificazione dell'immagine](#)
- [Esempi di file specifici per Amazon Linux](#)

Il file `/etc/system-release`

Amazon Linux include un file `/etc/system-release` che specifica la versione corrente installata. Questo file viene aggiornato utilizzando i gestori di pacchetti e su Amazon Linux fa parte del `system-release` pacchetto. Anche se alcune altre distribuzioni come Fedora hanno questo file, esso non è presente nelle distribuzioni basate su Debian come Ubuntu.

Note

Il `/etc/system-release` file contiene una stringa leggibile dall'uomo e non deve essere usato a livello di codice per identificare un sistema operativo o una versione. Utilizza invece i campi leggibili dalla macchina in `/etc/os-release` (o se non esiste). `/usr/lib/os-release` `/etc/os-release`

Amazon Linux contiene anche una versione leggibile dalla macchina `/etc/system-release` che segue la specifica Common Platform Enumeration (CPE) nel file. `/etc/system-release-cpe`

File di identificazione dell'immagine

Ogni immagine Amazon Linux contiene un `/etc/image-id` file univoco che fornisce informazioni aggiuntive sull'immagine originale generata dal team di Amazon Linux. Questo file è specifico di Amazon Linux e non si trova in altre distribuzioni Linux come Debian, Ubuntu o Fedora. Questo file contiene le seguenti informazioni relative all'immagine:

- `image_name, image_version, image_arch` — Valori della ricetta di compilazione utilizzata per costruire l'immagine.
- `image_stamp`: un valore esadecimale casuale univoco generato durante la creazione dell'immagine.
- `image_date`— L'ora UTC di creazione dell'immagine, in YYYYMMDDhhmmss formato.
- `recipe_name, recipe_id` — Il nome e l'ID della ricetta di compilazione utilizzata per costruire l'immagine.

Esempi di file specifici per Amazon Linux

Le seguenti sezioni forniscono esempi di file di identificazione specifici di Amazon Linux per ogni versione principale di Amazon Linux.

Note

In qualsiasi codice reale, `/usr/lib/os-release` dovrebbe essere usato se il `/etc/os-release` file non esiste.

AL2023

Gli esempi seguenti mostrano i file di identificazione per AL2 023.

Esempio di `/etc/image-id` AL2 023:

```
[ec2-user ~]$ cat /etc/image-id
```

```
image_name="al2023-container"  
image_version="2023"  
image_arch="x86_64"  
image_file="al2023-container-2023.8.20250721.2-x86_64"  
image_stamp="822b-1a9e"  
image_date="20250719211531"  
recipe_name="al2023 container"  
recipe_id="89b25f7b-be82-2215-a8eb-6e63-0830-94ea-658d41c4"
```

Esempio di `/etc/system-release` AL2 023:

```
[ec2-user ~]$ cat /etc/system-release
```

```
Amazon Linux release 2023.8.20250721 (Amazon Linux)
```

AL2

Gli esempi seguenti mostrano i file di identificazione per AL2.

Esempio di `/etc/image-id` per AL2:

```
[ec2-user ~]$ cat /etc/image-id
```

```
image_name="amzn2-container-raw"  
image_version="2"  
image_arch="x86_64"
```

```
image_file="amzn2-container-raw-2.0.20250721.2-x86_64"  
image_stamp="4126-16ad"  
image_date="20250721225801"  
recipe_name="amzn2 container"  
recipe_id="948422df-a4e6-5fc8-ba89-ef2e-0e1f-e1bb-16f84087"
```

Esempio di `/etc/system-release` per AL2:

```
[ec2-user ~]$ cat /etc/system-release
```

```
Amazon Linux release 2 (Karoo)
```

Amazon Linux AMI

Gli esempi seguenti mostrano i file di identificazione per l'AMI Amazon Linux.

Esempio `/etc/image-id` di AMI Amazon Linux:

```
[ec2-user ~]$ cat /etc/image-id
```

```
image_name="amzn-container-minimal"  
image_version="2018.03"  
image_arch="x86_64"  
image_file="amzn-container-minimal-2018.03.0.20231218.0-x86_64"  
image_stamp="407d-5ef3"  
image_date="20231218203210"  
recipe_name="amzn container"  
recipe_id="b1e7635e-14e3-dd57-b1ab-7351-edd0-d9e0-ca6852ea"
```

Esempio `/etc/system-release` di AMI Amazon Linux:

```
[ec2-user ~]$ cat /etc/system-release
```

```
Amazon Linux AMI release 2018.03
```

Codice di esempio per il rilevamento del sistema operativo

Gli esempi seguenti mostrano come rilevare a livello di codice il sistema operativo e la versione utilizzando il file `/etc/os-release` (o `/usr/lib/os-release` se `/etc/os-release` non

esiste). Questi esempi mostrano come distinguere tra Amazon Linux e altre distribuzioni, nonché come utilizzare il `ID_LIKE` campo per determinare le famiglie di distribuzione.

Lo script seguente è implementato in diversi linguaggi di programmazione e ogni implementazione produrrà lo stesso risultato.

Shell

```
#!/bin/bash

# Function to get a specific field from os-release file
get_os_release_field() {
    local field="$1"
    local os_release_file

    # Find the os-release file
    if [ -f /etc/os-release ]; then
        os_release_file='/etc/os-release'
    elif [ -f /usr/lib/os-release ]; then
        os_release_file='/usr/lib/os-release'
    else
        echo "Error: os-release file not found" >&2
        return 1
    fi

    # Source the file in a subshell and return the requested field.
    #
    # A subshell means that variables from os-release are only available
    # within the subshell, and the main script environment remains clean.
    (
        . "$os_release_file"
        eval "echo \"\${$field}\""
    )
}

is_amazon_linux() {
    [ "$(get_os_release_field ID)" = "amzn" ]
}

is_fedora() {
    [ "$(get_os_release_field ID)" = "fedora" ]
}
```

```
is_ubuntu() {
    [ "$(get_os_release_field ID)" = "ubuntu" ]
}

is_debian() {
    [ "$(get_os_release_field ID)" = "debian" ]
}

# Function to check if this is like Fedora (includes Amazon Linux, CentOS, RHEL,
etc.)
is_like_fedora() {
    local id="$(get_os_release_field ID)"
    local id_like="$(get_os_release_field ID_LIKE)"
    [ "$id" = "fedora" ] || [[ "$id_like" == *"fedora"* ]]
}

# Function to check if this is like Debian (includes Ubuntu and derivatives)
is_like_debian() {
    local id="$(get_os_release_field ID)"
    local id_like="$(get_os_release_field ID_LIKE)"
    [ "$id" = "debian" ] || [[ "$id_like" == *"debian"* ]]
}

# Get the main fields we'll use multiple times
ID="$(get_os_release_field ID)"
VERSION_ID="$(get_os_release_field VERSION_ID)"
PRETTY_NAME="$(get_os_release_field PRETTY_NAME)"
ID_LIKE="$(get_os_release_field ID_LIKE)"

echo "Operating System Detection Results:"
echo "====="
echo "Is Amazon Linux: $(is_amazon_linux && echo YES || echo NO)"
echo "Is Fedora: $(is_fedora && echo YES || echo NO)"
echo "Is Ubuntu: $(is_ubuntu && echo YES || echo NO)"
echo "Is Debian: $(is_debian && echo YES || echo NO)"
echo "Is like Fedora: $(is_like_fedora && echo YES || echo NO)"
echo "Is like Debian: $(is_like_debian && echo YES || echo NO)"
echo
echo "Detailed OS Information:"
echo "====="
echo "ID: $ID"
echo "VERSION_ID: $VERSION_ID"
echo "PRETTY_NAME: $PRETTY_NAME"
[ -n "$ID_LIKE" ] && echo "ID_LIKE: $ID_LIKE"
```

```
# Amazon Linux specific information
if is_amazon_linux; then
    echo ""
    echo "Amazon Linux Version Details:"
    echo "======"
    case "$VERSION_ID" in
        2018.03)
            echo "Amazon Linux AMI (version 1)"
            ;;
        2)
            echo "Amazon Linux 2"
            ;;
        2023)
            echo "Amazon Linux 2023"
            ;;
        *)
            echo "Unknown Amazon Linux version: $VERSION_ID"
            ;;
    esac

    # Check for Amazon Linux specific files
    [ -f /etc/image-id ] && echo "Amazon Linux image-id file present"
fi
```

Python 3.7-3.9

```
#!/usr/bin/env python3

import os
import sys

def parse_os_release():
    """Parse the os-release file and return a dictionary of key-value pairs."""
    os_release_data = {}

    # Try /etc/os-release first, then /usr/lib/os-release
    for path in ['/etc/os-release', '/usr/lib/os-release']:
        if os.path.exists(path):
            try:
                with open(path, 'r') as f:
                    for line in f:
                        line = line.strip()
```

```
        if line and not line.startswith('#') and '=' in line:
            key, value = line.split('=', 1)
            # Remove quotes if present
            value = value.strip('"\'')
            os_release_data[key] = value

    return os_release_data
except IOError:
    continue

print("Error: os-release file not found")
sys.exit(1)

def is_amazon_linux(os_data):
    """Check if this is Amazon Linux."""
    return os_data.get('ID') == 'amzn'

def is_fedora(os_data):
    """Check if this is Fedora."""
    return os_data.get('ID') == 'fedora'

def is_ubuntu(os_data):
    """Check if this is Ubuntu."""
    return os_data.get('ID') == 'ubuntu'

def is_debian(os_data):
    """Check if this is Debian."""
    return os_data.get('ID') == 'debian'

def is_like_fedora(os_data):
    """Check if this is like Fedora (includes Amazon Linux, CentOS, RHEL, etc.)."""
    if os_data.get('ID') == 'fedora':
        return True
    id_like = os_data.get('ID_LIKE', '')
    return 'fedora' in id_like

def is_like_debian(os_data):
    """Check if this is like Debian (includes Ubuntu and derivatives)."""
    if os_data.get('ID') == 'debian':
        return True
    id_like = os_data.get('ID_LIKE', '')
    return 'debian' in id_like

def main():
    # Parse os-release file
```

```
os_data = parse_os_release()

# Display results
print("Operating System Detection Results:")
print("=====")
print(f"Is Amazon Linux: {'YES' if is_amazon_linux(os_data) else 'NO'}")
print(f"Is Fedora: {'YES' if is_fedora(os_data) else 'NO'}")
print(f"Is Ubuntu: {'YES' if is_ubuntu(os_data) else 'NO'}")
print(f"Is Debian: {'YES' if is_debian(os_data) else 'NO'}")
print(f"Is like Fedora: {'YES' if is_like_fedora(os_data) else 'NO'}")
print(f"Is like Debian: {'YES' if is_like_debian(os_data) else 'NO'}")

# Additional information
print()
print("Detailed OS Information:")
print("=====")
print(f"ID: {os_data.get('ID', '')}")
print(f"VERSION_ID: {os_data.get('VERSION_ID', '')}")
print(f"PRETTY_NAME: {os_data.get('PRETTY_NAME', '')}")
if os_data.get('ID_LIKE'):
    print(f"ID_LIKE: {os_data.get('ID_LIKE')}")

# Amazon Linux specific information
if is_amazon_linux(os_data):
    print()
    print("Amazon Linux Version Details:")
    print("=====")
    version_id = os_data.get('VERSION_ID', '')
    if version_id == '2018.03':
        print("Amazon Linux AMI (version 1)")
    elif version_id == '2':
        print("Amazon Linux 2")
    elif version_id == '2023':
        print("Amazon Linux 2023")
    else:
        print(f"Unknown Amazon Linux version: {version_id}")

# Check for Amazon Linux specific files
if os.path.exists('/etc/image-id'):
    print("Amazon Linux image-id file present")

if __name__ == '__main__':
    main()
```

Python 3.10+

```
#!/usr/bin/env python3

import os
import sys
import platform

def is_amazon_linux(os_data):
    """Check if this is Amazon Linux."""
    return os_data.get('ID') == 'amzn'

def is_fedora(os_data):
    """Check if this is Fedora."""
    return os_data.get('ID') == 'fedora'

def is_ubuntu(os_data):
    """Check if this is Ubuntu."""
    return os_data.get('ID') == 'ubuntu'

def is_debian(os_data):
    """Check if this is Debian."""
    return os_data.get('ID') == 'debian'

def is_like_fedora(os_data):
    """Check if this is like Fedora (includes Amazon Linux, CentOS, RHEL, etc.)."""
    if os_data.get('ID') == 'fedora':
        return True
    id_like = os_data.get('ID_LIKE', '')
    return 'fedora' in id_like

def is_like_debian(os_data):
    """Check if this is like Debian (includes Ubuntu and derivatives)."""
    if os_data.get('ID') == 'debian':
        return True
    id_like = os_data.get('ID_LIKE', '')
    return 'debian' in id_like

def main():
    # Parse os-release file using the standard library function (Python 3.10+)
    try:
        os_data = platform.freedesktop_os_release()
    except OSError:
        print("Error: os-release file not found")
```

```
sys.exit(1)

# Display results
print("Operating System Detection Results:")
print("=====")
print(f"Is Amazon Linux: {'YES' if is_amazon_linux(os_data) else 'NO'}")
print(f"Is Fedora: {'YES' if is_fedora(os_data) else 'NO'}")
print(f"Is Ubuntu: {'YES' if is_ubuntu(os_data) else 'NO'}")
print(f"Is Debian: {'YES' if is_debian(os_data) else 'NO'}")
print(f"Is like Fedora: {'YES' if is_like_fedora(os_data) else 'NO'}")
print(f"Is like Debian: {'YES' if is_like_debian(os_data) else 'NO'}")

# Additional information
print()
print("Detailed OS Information:")
print("=====")
print(f"ID: {os_data.get('ID', '')}")
print(f"VERSION_ID: {os_data.get('VERSION_ID', '')}")
print(f"PRETTY_NAME: {os_data.get('PRETTY_NAME', '')}")
if os_data.get('ID_LIKE'):
    print(f"ID_LIKE: {os_data.get('ID_LIKE')}")

# Amazon Linux specific information
if is_amazon_linux(os_data):
    print()
    print("Amazon Linux Version Details:")
    print("=====")
    version_id = os_data.get('VERSION_ID', '')
    if version_id == '2018.03':
        print("Amazon Linux AMI (version 1)")
    elif version_id == '2':
        print("Amazon Linux 2")
    elif version_id == '2023':
        print("Amazon Linux 2023")
    else:
        print(f"Unknown Amazon Linux version: {version_id}")

# Check for Amazon Linux specific files
if os.path.exists('/etc/image-id'):
    print("Amazon Linux image-id file present")

if __name__ == '__main__':
    main()
```

Perl

```
#!/usr/bin/env perl

use strict;
use warnings;

# Function to parse the os-release file and return a hash of key-value pairs
sub parse_os_release {
    my %os_release_data;

    # Try /etc/os-release first, then /usr/lib/os-release
    my @paths = ('/etc/os-release', '/usr/lib/os-release');

    for my $path (@paths) {
        if (-f $path) {
            if (open(my $fh, '<', $path)) {
                while (my $line = <$fh>) {
                    chomp $line;
                    next if $line =~ /\s*$/ || $line =~ /\s*#/;

                    if ($line =~ /^(([^\=]+)=(.*)$/)) {
                        my ($key, $value) = ($1, $2);
                        # Remove quotes if present
                        $value =~ s/^[\"']|['\"]$/g;
                        $os_release_data{$key} = $value;
                    }
                }
                close($fh);
                return %os_release_data;
            }
        }
    }

    die "Error: os-release file not found\n";
}

# Function to check if this is Amazon Linux
sub is_amazon_linux {
    my %os_data = @_;
    return ($os_data{ID} // '') eq 'amzn';
}

# Function to check if this is Fedora
```

```
sub is_fedora {
    my %os_data = @_;
    return ($os_data{ID} // '') eq 'fedora';
}

# Function to check if this is Ubuntu
sub is_ubuntu {
    my %os_data = @_;
    return ($os_data{ID} // '') eq 'ubuntu';
}

# Function to check if this is Debian
sub is_debian {
    my %os_data = @_;
    return ($os_data{ID} // '') eq 'debian';
}

# Function to check if this is like Fedora (includes Amazon Linux, CentOS, RHEL,
etc.)
sub is_like_fedora {
    my %os_data = @_;
    return 1 if ($os_data{ID} // '') eq 'fedora';
    my $id_like = $os_data{ID_LIKE} // '';
    return $id_like =~ /fedora/;
}

# Function to check if this is like Debian (includes Ubuntu and derivatives)
sub is_like_debian {
    my %os_data = @_;
    return 1 if ($os_data{ID} // '') eq 'debian';
    my $id_like = $os_data{ID_LIKE} // '';
    return $id_like =~ /debian/;
}

# Main execution
my %os_data = parse_os_release();

# Display results
print "Operating System Detection Results:\n";
print "=====\n";
print "Is Amazon Linux: " . (is_amazon_linux(%os_data) ? "YES" : "NO") . "\n";
print "Is Fedora: " . (is_fedora(%os_data) ? "YES" : "NO") . "\n";
print "Is Ubuntu: " . (is_ubuntu(%os_data) ? "YES" : "NO") . "\n";
print "Is Debian: " . (is_debian(%os_data) ? "YES" : "NO") . "\n";
```

```

print "Is like Fedora: " . (is_like_fedora(%os_data) ? "YES" : "NO") . "\n";
print "Is like Debian: " . (is_like_debian(%os_data) ? "YES" : "NO") . "\n";
print "\n";

# Additional information
print "Detailed OS Information:\n";
print "=====\n";
print "ID: " . ($os_data{ID} // '') . "\n";
print "VERSION_ID: " . ($os_data{VERSION_ID} // '') . "\n";
print "PRETTY_NAME: " . ($os_data{PRETTY_NAME} // '') . "\n";
print "ID_LIKE: " . ($os_data{ID_LIKE} // '') . "\n" if $os_data{ID_LIKE};

# Amazon Linux specific information
if (is_amazon_linux(%os_data)) {
    print "\n";
    print "Amazon Linux Version Details:\n";
    print "=====\n";
    my $version_id = $os_data{VERSION_ID} // '';

    if ($version_id eq '2018.03') {
        print "Amazon Linux AMI (version 1)\n";
    } elsif ($version_id eq '2') {
        print "Amazon Linux 2\n";
    } elsif ($version_id eq '2023') {
        print "Amazon Linux 2023\n";
    } else {
        print "Unknown Amazon Linux version: $version_id\n";
    }

    # Check for Amazon Linux specific files
    if (-f '/etc/image-id') {
        print "Amazon Linux image-id file present\n";
    }
}

```

Quando viene eseguito su sistemi diversi, lo script produrrà il seguente output:

AL2023

```

Operating System Detection Results:
=====
Is Amazon Linux: YES

```

```
Is Fedora: NO
Is Ubuntu: NO
Is Debian: NO
Is like Fedora: YES
Is like Debian: NO

Detailed OS Information:
=====
ID: amzn
VERSION_ID: 2023
PRETTY_NAME: Amazon Linux 2023.8.20250721
ID_LIKE: fedora

Amazon Linux Version Details:
=====
Amazon Linux 2023
Amazon Linux image-id file present
```

AL2

```
Operating System Detection Results:
=====
Is Amazon Linux: YES
Is Fedora: NO
Is Ubuntu: NO
Is Debian: NO
Is like Fedora: YES
Is like Debian: NO

Detailed OS Information:
=====
ID: amzn
VERSION_ID: 2
PRETTY_NAME: Amazon Linux 2
ID_LIKE: centos rhel fedora

Amazon Linux Version Details:
=====
Amazon Linux 2
Amazon Linux image-id file present
```

Amazon Linux AMI

Operating System Detection Results:

=====

```
Is Amazon Linux: YES
Is Fedora: NO
Is Ubuntu: NO
Is Debian: NO
Is like Fedora: YES
Is like Debian: NO
```

Detailed OS Information:

=====

```
ID: amzn
VERSION_ID: 2018.03
PRETTY_NAME: Amazon Linux AMI 2018.03
ID_LIKE: rhel fedora
```

Amazon Linux Version Details:

=====

```
Amazon Linux AMI (version 1)
Amazon Linux image-id file present
```

Ubuntu

Operating System Detection Results:

=====

```
Is Amazon Linux: NO
Is Fedora: NO
Is Ubuntu: YES
Is Debian: NO
Is like Fedora: NO
Is like Debian: YES
```

Detailed OS Information:

=====

```
ID: ubuntu
VERSION_ID: 24.04
PRETTY_NAME: Ubuntu 24.04.2 LTS
ID_LIKE: debian
```

Debian

```
Operating System Detection Results:
=====
Is Amazon Linux: NO
Is Fedora: NO
Is Ubuntu: NO
Is Debian: YES
Is like Fedora: NO
Is like Debian: YES

Detailed OS Information:
=====
ID: debian
VERSION_ID: 12
PRETTY_NAME: Debian GNU/Linux 12 (bookworm)
```

Fedora

```
Operating System Detection Results:
=====
Is Amazon Linux: NO
Is Fedora: YES
Is Ubuntu: NO
Is Debian: NO
Is like Fedora: YES
Is like Debian: NO

Detailed OS Information:
=====
ID: fedora
VERSION_ID: 42
PRETTY_NAME: Fedora Linux 42 (Container Image)
```

Layout del file system

Questa sezione tratta il layout del file system di un sistema AL2 023, inclusi dettagli che possono essere specifici delle istanze o dei contenitori basati su 023. AL2 Vedi il `file-hierarchy(7)` man pagina per ulteriori informazioni.

Argomenti

- [/\(La directory principale\)](#)
- [/boot\(Kernel, initramfs, ecc.\)](#)
- [/etc\(Configurazione del sistema\)](#)
- [/home\(Le home directory degli utenti\)](#)
- [/root \(root home directory dell'utente\)](#)
- [/srv\(Payload del server\)](#)
- [/tmp\(file temporanei di piccole dimensioni\)](#)
- [/run\(dati di runtime\)](#)
- [/usr\(Risorse di sistema\)](#)
- [/var\(Dati di sistema variabili persistenti\)](#)

/(La directory principale)

Per impostazione predefinita, le immagini AL2 023 sono configurate con un file scrivibile/, che consente agli utenti privilegiati di creare nuovi file e directory.

È possibile configurare `systemd` i servizi in modo che utilizzino un percorso o un'immagine diversi in modo che appaiano come / per quel servizio, nonché imporre restrizioni di accesso su qualsiasi percorso.

Note

È consigliabile configurare i `systemd` servizi in modo da limitare l'accesso a cui il servizio ha accesso. Ciò può includere l'utilizzo della `ReadOnlyPaths=` direttiva che rende la sola / lettura per quel servizio.

Per ulteriori informazioni sull'utilizzo `systemd` per limitare l'accesso di un servizio al sistema, consulta la `systemd.exec(5)` man pagina.

/boot(Kernel, initramfs, ecc.)

Per impostazione predefinita, le immagini AL2 023 avviabili sono configurate in modo da trovarsi su `/boot` root file system. Il `/boot` percorso è rilevante solo per le immagini avviabili, quindi non è utilizzato nelle immagini del contenitore AL2 023.

Questa directory contiene i file necessari per l'avvio di AL2 023, come il kernel Linux e `initramfs`. Il contenuto di questa directory deve essere manipolato solo utilizzando gli strumenti forniti con il sistema operativo.

/boot/efi(Partizione di sistema EFI)

Per impostazione predefinita, le immagini AL2 023 avviabili sono configurate con EFI System partizione in cui si sta montando. `/boot/efi` Questo file system è gestito dal sistema operativo e contiene codice e configurazioni fondamentali per l'avvio del sistema.

Questo percorso non è rilevante per le immagini dei contenitori.

/etc(Configurazione del sistema)

La `/etc` directory su AL2 023 contiene la configurazione specifica del sistema. Per impostazione predefinita, AL2 023 immagini vengono fornite con `/etc` root filesystem e scrivibile da utenti privilegiati.

Note

È normale che le applicazioni (inclusesystemd) mantengano la configurazione predefinita in base alla [/usr\(Risorse di sistema\)](#) quale è possibile sovrascrivere inserendo la configurazione. [/etc\(Configurazione del sistema\)](#)

Per queste applicazioni, la modifica dei file [/usr\(Risorse di sistema\)](#) anziché sovrascrivere la configurazione predefinita `/etc` comporterà probabilmente la sovrascrittura delle modifiche quando il pacchetto viene aggiornato.

/home(Le home directory degli utenti)

Gli utenti normali hanno le loro home directory sotto `/home`, ma il software dovrebbe sempre cercare la variabile di `$HOME` ambiente per utente piuttosto che basarsi su uno schema come `/home/$USER`

Per impostazione predefinita, sono presenti AL2 023 immagini /home root file system, ma il software non dovrebbe basarsi su questo. È perfettamente valido che il sistema operativo sia configurato come un file system separato, che viene montato successivamente durante l'avvio o solo dopo l'autenticazione dell'utente nel sistema. /home

Il root la home directory dell'utente non è presente/home, ma lo è in [/root \(root home directory dell'utente\)](#) modo che sia disponibile nel caso in cui il /home file system non possa essere montato.

Note

È consigliabile che i systemd servizi che non richiedono l'/homeaccesso in scrittura siano configurati con la `ProtectHome=read-only` direttiva. Con questa opzione, `/home/root`, e `/run/user` vengono resi di sola lettura per il servizio.

È inoltre consigliabile che i servizi che non richiedono alcun accesso /home siano configurati con la `ProtectHome=tmpfs` direttiva, che eseguirà il servizio in una sandbox dove `/home/root`, e `/run/user` sono file system vuoti di sola letturatmpfs.

Per ulteriori informazioni sull'utilizzo systemd per limitare l'accesso di un servizio al sistema, consulta la `systemd.exec(5)` man pagina.

/root (root home directory dell'utente)

La home directory di root utente è la /root directory, volutamente separata da essa in [/home\(Le home directory degli utenti\)](#) modo che sia presente nel caso in cui si trovi su un file system che [/home\(Le home directory degli utenti\)](#) non è disponibile.

La procedura ottimale per la configurazione systemd dei servizi è la stessa di quella per/root. [/home\(Le home directory degli utenti\)](#)

/srv(Payload del server)

La /srv directory è gestita dall'amministratore del sistema e Amazon Linux 2023 non impone restrizioni sull'organizzazione di questa directory.

È possibile configurare la /srv directory in modo che si trovi su un file system separato, quindi potrebbe essere disponibile solo in seguito all'avvio.

/tmp(file temporanei di piccole dimensioni)

Note

Amazon Linux 2023 è diverso da Amazon Linux 2 in quanto, per impostazione predefinita, lo /tmp è ora tmpfs piuttosto che un percorso sul root file system.

Note

Quando si esegue in un contenitore, in genere è la configurazione di runtime del contenitore a determinare se lo /tmp è tmpfs o il percorso sul disco e se è in corso o meno un processo di pulizia.

La /tmp directory è per file temporanei di piccole dimensioni limitate. Per impostazione predefinita, AL2 023 la configura come un tmpfs file system con un limite di dimensione del 50% della RAM e un massimo di un milione inodes.

Le applicazioni dovrebbero preferire il percorso nella variabile di \$TMPDIR ambiente rispetto a. /tmp. Gli utenti possono quindi impostare la variabile di \$TMPDIR ambiente per sovrascrivere il percorso per cui un'applicazione deve utilizzare /tmp.

Per file temporanei più grandi, [/var/tmp](#) dovrebbe invece essere utilizzata.

Warning

Poiché /tmp è condiviso, è importante utilizzare metodi sicuri per creare file temporanei. Per i dettagli, consulta la systemd documentazione originale su [Using /tmp and /var/tmp Safely](#).

Note

È consigliabile che i systemd servizi siano configurati con la PrivateTmp= direttiva impostata su yes o disconnected che eseguano il servizio in una sandbox dove /tmp e non [/var/tmp](#) siano condivisi con l'host o con altri servizi.

Per ulteriori informazioni, incluso come configurare due servizi per condividere le stesse directory temporanee private, consulta `systemd.exec(5)` man pagina.

Il contenuto di `/tmp` viene in genere pulito all'avvio e i file non utilizzati vengono puliti regolarmente. Per impostazione predefinita, il processo di pulizia viene eseguito subito dopo l'avvio e poi ogni giorno. Per informazioni su come configurare la pulizia dei file temporanei, consulta e `tmpfiles.d(5)` `systemd-tmpfiles(8)` man pagine man.

I [/var/tmp](#) percorsi `/tmp` e sono strettamente correlati ed esistono per scopi diversi.

/run(dati di runtime)

La `/run` directory viene utilizzata dai pacchetti di sistema per memorizzare piccole quantità di dati di runtime (come i file socket). È un tmpfs file system ed è scrivibile solo da programmi privilegiati.

La `/run/log` directory può essere utilizzata dai componenti del sistema per memorizzare i log in, prima di essere scritti `/var/log` o prima che il `/var/log` file system sia disponibile.

Il `/run/user/` percorso contiene le directory di runtime per utente. Per impostazione predefinita, si tratta di singoli tmpfs file system montati al `systemd` momento dell'accesso dell'utente e cancellati quando l'utente non è più connesso. Secondo la [XDG Base Directory Specification](#), questi percorsi non devono essere referenziati direttamente ma piuttosto tramite la variabile di ambiente. `$XDG_RUNTIME_DIR`

/usr(Risorse di sistema)

La `/usr` gerarchia riguarda le risorse del sistema operativo fornite dal fornitore. Ad eccezione della [/usr/local](#) gerarchia, nulla dovrebbe modificare nulla `/usr` tranne il gestore di pacchetti del sistema operativo.

Le applicazioni software devono presupporre che `/usr` possano essere di sola lettura. La `/usr` gerarchia non deve essere utilizzata per dati volitivi. Ad eccezione di [/usr/local](#), la `/usr` gerarchia non deve essere utilizzata per i dati aggiunti o modificati al di fuori dell'installazione/rimozione dei pacchetti, come avviene dal gestore di pacchetti del sistema operativo. Il gestore di pacchetti del sistema operativo può presumere che tutta la `/usr` gerarchia (eccetto [/usr/local](#)) sia lo stesso punto di montaggio.

Il software installato al di fuori del gestore di pacchetti del sistema operativo non deve archiviare dati in `/usr` quanto ciò potrebbe impedire future invocazioni del gestore di pacchetti del sistema operativo. La `/usr/local` gerarchia è l'eccezione ed è riservata al software esterno al gestore di pacchetti del sistema operativo.

`/usr/bin`(Eseguibili)

File eseguibili che dovrebbero apparire nella ricerca `$PATH` standard e sono utili da richiamare da una shell. I demoni e gli eseguibili che non è utile invocare da una shell risiedono invece in `or. /usr/lib /usr/libexec`

`/usr/include`(Intestazioni C/C++)

La `/usr/include` directory contiene i file di intestazione C e C++, generalmente contenuti in pacchetti con il suffisso. `-devel`

`/usr/libe /usr/lib64` (Librerie condivise)

Su Amazon Linux 2023, il `/usr/lib64` percorso viene utilizzato per le librerie condivise a 64 bit e i dati dei pacchetti che dipendono dall'architettura. Poiché AL2 023 non include alcun supporto per lo spazio utente a 32 bit, sono disponibili solo librerie condivise a 64 bit.

Il `/usr/lib` percorso è per i dati statici provenienti dai pacchetti del sistema operativo ed è compatibile con tutte le architetture. Ciò può includere eseguibili che di solito non vengono richiamati da una shell, che possono essere trovati anche in `/usr/libexec` Le librerie condivise si trovano in `/usr/lib64` anziché `/usr/lib`

`/usr/local`(Software installato dall'amministratore di sistema)

Su Amazon Linux 2023, l'`/usr/local` amministratore di sistema può installare il software in un software che non è di proprietà del sistema operativo e che non verrà toccato dal sistema operativo. La `/usr/local` gerarchia predefinita rispecchia la gerarchia. `/`

`/usr/share`(Risorse condivise)

Risorse condivise come documentazione, caratteri e dati sul fuso orario sono disponibili. `/usr/share` È comune che diverse specifiche stabiliscano esattamente dove e in quale formato i dati vengono archiviati in questa directory.

/usr/share/doc(Documentazione)

La documentazione fornita con i pacchetti verrà archiviata in `/usr/share/doc`.

/var(Dati di sistema variabili persistenti)

/var/cache(Cache)

Al contrario [/var/lib](#), la cancellazione dei dati non `/var/cache` comporterà la perdita di dati, poiché le applicazioni devono essere in grado di ricostruire `/var/cache` i propri dati da altre fonti.

/var/lib(Dati di sistema persistenti)

La `/var/lib` directory viene utilizzata per i dati di sistema persistenti. Vari componenti del sistema inseriranno qui i dati che sono privati di quel componente. Al contrario [/var/cache](#), la cancellazione dei dati `/var/lib` comporterà la perdita di dati.

Ad esempio, per impostazione predefinita, il server del database PostgreSQL memorizzerà i dati del database in `/var/lib/pgsql`. Il layout e i formati di file di questi dati sono privati per PostgreSQL e si tratta di dati persistenti, poiché se cancellati, l'utente subisce una perdita di dati.

/var/log(Registri persistenti)

Questa directory viene utilizzata per archiviare i log persistenti. È consigliabile che il software utilizzi le chiamate `syslog(3)` o `sd_journal_print(3)` API anziché archiviare direttamente i file di registro in `/var/log`.

Note

In AL2 023 il [systemd](#) diario [sostituisce rsyslog](#), che è una notevole differenza rispetto alla configurazione predefinita di Amazon Linux 2.

Per ulteriori informazioni sulla lettura dei log utilizzando `journalctl`, consulta la [journalctl](#) pagina di manuale.

Molte applicazioni utilizzano meccanismi propri per scrivere, e talvolta ruotare, i file di registro presenti in `/var/log`. Consultate la documentazione di queste applicazioni su come configurare i relativi file di registro.

/var/spool(Code di posta e stampante)

Questa directory viene utilizzata per dati persistenti come le code di posta o di stampa.

/var/tmp(file temporanei più grandi)

Al suo posto [/tmp](#) dovrebbe essere eventualmente utilizzata per file temporanei di piccole dimensioni.

Per impostazione predefinita, mentre [/tmp](#) è configurato come tmpfs volume, `/var/tmp` è configurato di default per essere un percorso sul file system principale, ed è quindi la soluzione ideale per file temporanei più grandi e persistenti. Per impostazione predefinita, viene eseguito regolarmente un processo di pulizia che rimuove i file a cui non si accede di recente.

Per informazioni su come configurare la pulizia dei file temporanei, consulta `tmpfiles.d(5)` `systemd-tmpfiles(8)` `man` pagine `man`.

Analogamente [/tmp](#), le applicazioni dovrebbero preferire il percorso specificato nella variabile di `$TMPDIR` ambiente rispetto a `/var/tmp`. Gli utenti possono quindi impostare la variabile di `$TMPDIR` ambiente per sovrascrivere il percorso per `/var/tmp` cui deve essere utilizzata un'applicazione.

Warning

Poiché `/var/tmp` è condivisa (così com'è) [/tmp](#), è importante utilizzare metodi sicuri per creare file temporanei. Per i dettagli, consulta la `systemd` documentazione originale su [Using /tmp and /var/tmp Safely](#).

Note

È consigliabile che i `systemd` servizi siano configurati con la `PrivateTmp=` direttiva impostata su `yes` o `disconnected` che eseguano il servizio in una sandbox dove [/tmp](#) e non [/var/tmp](#) siano condivisi con l'host o con altri servizi. Per ulteriori informazioni, incluso come configurare due servizi per condividere le stesse directory temporanee private, consulta `systemd.exec(5)` `man` pagina.

I [/var/tmp](#) percorsi [/tmp](#) e sono strettamente correlati ed esistono per scopi diversi.

Aggiornamento AL2023

È importante tenersi aggiornati sulle AL2023 versioni in modo da poter beneficiare degli aggiornamenti di sicurezza e delle nuove funzionalità. Con AL2023, puoi garantire la coerenza tra le versioni dei pacchetti e gli aggiornamenti in tutto l'ambiente tramite [Aggiornamenti deterministici tramite repository con versioni su AL2023](#).

Warning

`dnf --releasever=latest update` L'esecuzione non è una procedura consigliata ed è probabile che comporti il primo test di un aggiornamento del sistema operativo in produzione. Invece di utilizzare `latest`, utilizza una versione di AL2023 rilascio specifica. Ciò garantisce l'implementazione delle stesse modifiche tra le istanze di produzione testate in precedenza. Ad esempio, `dnf --releasever=2023.11.20260427 update` verrà sempre aggiornato alla versione 2023.11.20260427.

[Per ulteriori informazioni, consulta la sezione Aggiornamento AL2023 nella Guida per l'utente. AL2023](#)

Argomenti

- [Procedure consigliate per la distribuzione sicura degli aggiornamenti](#)
- [Ricevi notifiche sui nuovi aggiornamenti](#)
- [Aggiornamenti deterministici tramite repository con versioni su AL2023](#)
- [Gestisci gli aggiornamenti dei pacchetti e del sistema operativo in AL2023](#)
- [Kernel Live Patching attivo AL2023](#)
- [Aggiornamento del kernel Linux su AL2023](#)

Procedure consigliate per la distribuzione sicura degli aggiornamenti

Amazon Linux 2023 (AL2023) dispone di diverse funzionalità progettate per facilitare la distribuzione sicura degli aggiornamenti nel sistema operativo e per essere in grado di sapere cosa è cambiato tra un aggiornamento e l'altro e, se necessario, di ripristinare facilmente la versione precedente. Questa

sezione esplora le lezioni apprese AWS da oltre un decennio di utilizzo interno ed esterno di Amazon Linux.

⚠ Warning

`dnf --releasever=latest update`L'esecuzione non è una procedura consigliata ed è probabile che un aggiornamento del sistema operativo venga testato per la prima volta in produzione.

Invece di utilizzarla `latest`, utilizzate una versione di AL2023 rilascio specifica. Questo assicura che stiate implementando le stesse modifiche su tutte le istanze di produzione testate in precedenza. Ad esempio, `dnf --releasever=2023.11.20260427 update` verrà sempre aggiornato alla versione 2023.11.20260427.

[Per ulteriori informazioni, consulta la sezione Aggiornamento AL2023 nella Guida per l'utente. AL2023](#)

Senza pianificare la sicurezza dell'implementazione degli aggiornamenti del sistema operativo, l'impatto di un'interazione negativa imprevista tra l'utente application/service e un aggiornamento del sistema operativo può essere notevolmente maggiore, fino a includere un'interruzione totale. Come per qualsiasi problema relativo al software, prima viene rilevato il problema, minore è l'impatto che può avere sugli utenti finali.

È importante non cadere nella trappola di credere a due cose che fundamentalmente non sono vere:

1. Il fornitore del sistema operativo non commetterà mai errori durante un aggiornamento del sistema operativo.
2. Il comportamento specifico o l'interfaccia verso il sistema operativo su cui fate affidamento corrisponde al comportamento e alle interfacce su cui il fornitore del sistema operativo considererebbe qualcosa su cui fare affidamento.

vale a dire che sia il fornitore del sistema operativo che l'utente concorderebbe sul fatto che si è verificato un problema con l'aggiornamento.

Non fate affidamento sulle buone intenzioni, ma predisponete sistemi per garantire che la sicurezza dell'implementazione includa qualsiasi aggiornamento del sistema operativo.

Non è consigliabile testare i nuovi aggiornamenti del sistema operativo distribuendoli in ambienti di produzione. È consigliabile considerare il sistema operativo come un'altra parte della distribuzione e

prendere in considerazione l'applicazione degli stessi meccanismi di sicurezza dell'implementazione che si ritengono adatti per qualsiasi altra modifica all'ambiente di produzione.

È consigliabile testare tutti gli aggiornamenti del sistema operativo prima di distribuirli sui sistemi di produzione. Durante la distribuzione, si consigliano implementazioni graduali combinate con un buon monitoraggio. Le implementazioni in fasi possono garantire che se si verifica un problema, anche se non immediato, l'impatto sia limitato a un sottoinsieme di un parco macchine e che l'ulteriore implementazione dell'aggiornamento possa essere interrotta durante ulteriori indagini e mitigazioni.

La mitigazione di qualsiasi impatto negativo derivante dall'aggiornamento del sistema operativo è spesso la prima priorità, seguita dalla risoluzione del problema, ovunque si trovi. Laddove l'introduzione di un aggiornamento del sistema operativo è correlata a un impatto negativo, la possibilità di tornare alla versione precedente del sistema operativo nota come valida è uno strumento potente.

Amazon Linux 2023 introduce [Aggiornamenti deterministici tramite repository con versioni](#) una nuova potente funzionalità per garantire che qualsiasi modifica alla versione del sistema operativo (o ai singoli pacchetti) sia ripetibile. Pertanto, se si verifica un problema durante il passaggio da una versione del sistema operativo all'altra, sono disponibili meccanismi semplici da usare per attenersi alla versione del sistema operativo funzionante nota mentre si cerca di risolvere il problema.

Inoltre AL2023, ogni volta che rilasciamo nuovi aggiornamenti del pacchetto, c'è una nuova versione a cui affidarsi e una nuova versione AMIs si blocca su quella versione. Le [note di AL2023 rilascio](#) riguardano le modifiche apportate a ciascuna versione e [Avvisi di sicurezza di Amazon Linux per AL2023](#) i problemi di sicurezza risolti negli aggiornamenti dei pacchetti.

[Ad esempio, se sei interessato dal problema presente nella versione 2023.6.20241028, puoi tornare immediatamente a utilizzare le immagini AMIs e i contenitori della versione precedente, 2023.6.20241010.](#) In questo caso, c'era un bug in un pacchetto che è stato corretto nella successiva versione [2023.6.20241031](#), ma chiunque fosse interessato poteva agire immediatamente per mitigarlo: basta usare le immagini precedenti. [Aggiornamenti deterministici tramite repository con versioni](#)

[Aggiornamenti deterministici tramite repository con versioni](#) garantisce inoltre che qualsiasi implementazione in corso di un aggiornamento del sistema operativo, in atto o mediante il lancio di immagini nuove AMIs o di contenitori, non sia influenzata dagli aggiornamenti del sistema operativo rilasciati successivamente.

[Per il nostro primo esempio, la flotta A è una flotta di grandi dimensioni che è a metà della distribuzione dell'aggiornamento dalla versione 2023.5.20241001 alla versione 2023.6.20241010](#)

[quando esce la versione 2023.6.20241028. Aggiornamenti deterministici tramite repository con versioni](#) significa che la distribuzione per la flotta A continua senza alcuna modifica agli aggiornamenti che sta applicando.

Lo scopo delle strategie di dispiegamento basate sulle ondate o su fasi, ad esempio impiegando prima l'1% della flotta, poi il 5%, il 10%, il 20%, il 40%, fino a raggiungere il 100%, è quello di poter testare una modifica in modo limitato prima di estenderla a un livello più ampio. Questo tipo di strategia di implementazione è generalmente considerata la migliore pratica per implementare qualsiasi modifica della produzione.

Con una strategia di implementazione basata sull'ondata e la flotta Un aggiornamento alla versione [2023.6.20241010](#) è in una fase in cui viene distribuito su più host contemporaneamente, il fatto che [2023.6.20241028](#) sia stato rilasciato non ha alcun impatto sull'implementazione in corso grazie all'utilizzo. [Aggiornamenti deterministici tramite repository con versioni](#)

Se la flotta B eseguisse una versione precedente, ad esempio [2023.5.20240708](#), e avesse iniziato a distribuire l'aggiornamento alla versione [2023.6.20241028](#) e la flotta B fosse interessata dal problema in quella versione, ciò verrebbe notato all'inizio della distribuzione. A quel punto, si può decidere se sospendere l'implementazione fino a quando non sarà disponibile una soluzione per quel problema o se nel frattempo avviare una distribuzione della stessa versione della flotta A, [2023.6.20241010](#), in modo che la flotta B riceva tutti gli aggiornamenti tra il [2023.5.20240708](#) e il [2023.6.20241010](#).

È importante notare che non eseguire tempestivamente gli aggiornamenti del sistema operativo può causare problemi. È probabile che i nuovi aggiornamenti contengano bug e correzioni di sicurezza che potrebbero essere rilevanti per l'ambiente in uso. Per ulteriori informazioni, consultare [Sicurezza e conformità in Amazon Linux 2023](#) e [Gestisci gli aggiornamenti dei pacchetti e del sistema operativo in AL2023](#).

È importante configurare i sistemi di distribuzione in modo da poter ricevere facilmente nuovi aggiornamenti del sistema operativo, testarli prima di implementarli in produzione e utilizzare meccanismi come le distribuzioni basate su ondate per ridurre al minimo qualsiasi impatto negativo. Per poter mitigare l'impatto negativo di un aggiornamento del sistema operativo, è importante sapere come fare in modo che i sistemi di distribuzione puntino a una versione precedente del sistema operativo nota come valida e, una volta risolto il problema, non rimanere più vincolati alla versione precedente nota come valida, ma passare a una nuova versione riconosciuta valida.

Preparazione per gli aggiornamenti minori

La preparazione di aggiornamenti più piccoli del sistema operativo, ad esempio una nuova release puntuale di, AL2023 è pensata per essere limitata a zero sforzi. Assicuratevi di leggere le [note di AL2023 rilascio](#) per eventuali modifiche imminenti.

Il [periodo di supporto di un pacchetto](#) che sta per scadere può comportare il passaggio a una versione più recente del runtime del linguaggio (ad esempio con [PHP nel AL2 2023](#)). È consigliabile prepararsi a tale evenienza in anticipo passando alle nuove versioni di runtime linguistiche con largo anticipo rispetto alla fine del periodo di supporto.

Per pacchetti come questi [pcr versione 1](#), c'è anche la possibilità di pianificare in anticipo e migrare qualsiasi codice alla versione sostitutiva, che in questo caso è la pcr versione 2. È buona norma farlo il prima possibile, in modo da avere il tempo necessario per eventuali intoppi.

Se non esiste una sostituzione diretta, ad esempio con [libdb Berkeley DB \(\)](#), potrebbe essere necessario effettuare una scelta in base al caso d'uso.

Preparazione per gli aggiornamenti principali

L'aggiornamento a una nuova versione principale di un sistema operativo è considerato quasi universalmente come qualcosa che richiede pianificazione, lavoro per adattarsi a funzionalità modificate o obsolete e anche test prima della distribuzione. Non è raro essere in grado di prepararsi per la prossima versione principale di Amazon Linux 2023 in modo più incrementale, ad esempio affrontando qualsiasi utilizzo di funzionalità obsolete o rimosse prima di procedere con il passaggio alla versione principale successiva.

Ad esempio, quando si passa da AL2 a AL2023, la lettura della [Funzionalità obsoleta e rimossa in AL2 AL2023](#) sezione può comportare una serie di piccoli e sicuri passaggi che possono essere eseguiti mentre si è ancora in uso per prepararsi. AL2 AL2023 Ad esempio, qualsiasi [Python 2.7 è stato sostituito con Python 3](#) utilizzo (al di fuori dell'uso del sistema operativo, ad esempio nel gestore di yum pacchetti) può essere migrato a Python 3 in preparazione all'uso. [Python nel AL2023](#) Se si utilizza [PHP](#), sia AL2 (tramite PHP 8.2 [AL2 Extra](#)) che PHP 8.2, e AL2023 quindi sia la migrazione della versione di PHP che la migrazione del sistema operativo, non devono avvenire contemporaneamente.

Durante l'utilizzo AL2023, è anche possibile prepararsi per la prossima versione principale di Amazon Linux 2023 oggi stesso, durante l'utilizzo AL2023. La [Obsoleto nel 023 AL2](#) sezione tratta le funzionalità e i pacchetti che sono obsoleti AL2023 e che devono essere rimossi.

Ad esempio, la migrazione di qualsiasi [System V init \(sysvinit\)](#) utilizzo residuo, ad esempio degli `init` script, all'`systemd` equivalente, vi preparerà per le future sfide e vi consentirà di utilizzare l'intero set di `systemd` funzionalità per monitorare il servizio, come e se riavviarlo, di quali altri servizi necessita e se è necessario applicare vincoli di risorse o autorizzazioni.

Per funzionalità come il supporto a 32 bit, la deprecazione può estendersi a più versioni principali del sistema operativo. Per i modelli a 32 bit, Amazon Linux 1 (AL1) è obsoleto [x86 a 32 bit \(i686\) AMIs](#), Amazon Linux 2 è obsoleto e Amazon Linux 2023 è [Pacchetti x86 \(i686\) a 32 bit](#) obsoleto. [Supporto per runtime x86 \(i686\) a 32 bit](#) La transizione da riguarda anche diverse versioni principali del sistema operativo. [IMDSv1](#) Resta inteso che alcuni clienti richiedono più tempo per adattarsi a questi tipi di modifiche, quindi c'è un ampio margine di manovra prima che la funzionalità non sia più disponibile in Amazon Linux 2023.

L'elenco delle funzionalità obsolete viene aggiornato nel corso della vita del sistema operativo ed è consigliabile tenersi aggiornati sulle modifiche apportate.

Ricevi notifiche sui nuovi aggiornamenti

Puoi ricevere notifiche ogni volta che viene rilasciata una nuova AMI AL2 023. Le notifiche vengono pubblicate con [Amazon SNS](#) utilizzando il seguente argomento.

```
arn:aws:sns:us-east-1:137112412989:amazon-linux-2023-ami-updates
```

I messaggi vengono pubblicati qui quando viene pubblicata una nuova AMI AL2 023. La versione dell'AMI sarà inclusa nel messaggio.

Questi messaggi possono essere ricevuti utilizzando diversi metodi. Si consiglia di utilizzare il seguente metodo.

1. Apri la [console Amazon SNS](#).
2. Nella barra di navigazione, modificalo in Stati Uniti orientali (Virginia settentrionale), se necessario. Regione AWS Devi selezionare la regione in cui la notifica SNS per la quale hai effettuato la sottoscrizione è stata creata.
3. Nel pannello di navigazione, scegli Sottoscrizioni, quindi Crea sottoscrizione.
4. Nella finestra di dialogo Create subscription (Crea sottoscrizione) eseguire le seguenti operazioni:
 - a. Per l'argomento ARN, copia e incolla il seguente Amazon Resource Name (ARN):
arn:aws:sns:us-east-1:137112412989:amazon-linux-2023-ami-updates
 - b. Per Protocollo, scegli E-mail.

- c. In Endpoint immetti l'indirizzo e-mail utilizzabile per ricevere le notifiche.
 - d. Scegli Create Subscription (Crea sottoscrizione).
5. Riceverai un'e-mail di conferma con oggetto "AWS Notifica - Conferma dell'abbonamento». Apri l'e-mail e seleziona Conferma sottoscrizione per completare la sottoscrizione.

Aggiornamenti deterministici tramite repository con versioni su AL2023

Note

Per impostazione predefinita, l' AL2023 istanza non riceve automaticamente aggiornamenti di sicurezza critici e importanti aggiuntivi al momento del lancio. L'istanza contiene inizialmente gli aggiornamenti disponibili nella versione AL2023 e nell'AMI scelta.

Controllo degli aggiornamenti ricevuti dai rilasci principali e secondari

Con AL2023, puoi garantire la coerenza tra le versioni dei pacchetti e gli aggiornamenti in tutto l'ambiente. Puoi anche garantire la coerenza per più istanze della stessa Amazon Machine Image (AMI). Con la funzionalità degli aggiornamenti deterministici tramite repository con controllo delle versioni di AL2023, attivata per impostazione predefinita, puoi applicare gli aggiornamenti in base a una pianificazione che soddisfi le tue esigenze specifiche.

Ogni volta che rilasciamo nuovi aggiornamenti del pacchetto, c'è una nuova versione a cui affidarsi e una nuova versione AMIs che si collega a quella versione.

AL2023 si blocca su una versione specifica del tuo repository. Questa funzionalità è supportata sia per le versioni principali che per quelle secondarie. L' AL2023 AMI, esposta tramite i nostri parametri SSM, è sempre la versione più recente. Contiene il maggior numero di up-to-date pacchetti e aggiornamenti, inclusi gli aggiornamenti di sicurezza critici e importanti.

Se avvii un'istanza da un'AMI esistente, gli aggiornamenti non vengono applicati automaticamente. Tutti i pacchetti aggiuntivi installati come parte del provisioning vengono mappati alla versione del repository dell'AMI esistente.

Con questa funzionalità, hai la responsabilità di garantire la coerenza tra le versioni dei pacchetti e degli aggiornamenti in tutto l'ambiente. Questo è particolarmente vero se stai avviando più istanze

dalla stessa AMI. Puoi applicare gli aggiornamenti in base a una pianificazione in grado di soddisfare le tue esigenze. Puoi anche applicare un set specifico di aggiornamenti al momento dell'avvio, poiché questi possono anche essere collegati a una versione specifica del repository.

Differenze tra aggiornamenti delle versioni principali e secondarie

Le versioni principali di AL2023 includono aggiornamenti su larga scala e potrebbero aggiungere, eliminare o aggiornare pacchetti. Per garantire la compatibilità, aggiorna l'istanza a una nuova versione principale solo dopo aver testato l'applicazione su tale versione.

Le versioni minori di AL2023 includono aggiornamenti di funzionalità e sicurezza, ma non includono modifiche ai pacchetti. Ciò garantisce che le funzionalità di Linux e l'API della libreria di sistema rimangano disponibili nelle nuove versioni. Non è necessario testare l'applicazione prima dell'aggiornamento.

Sapere quando sono disponibili gli aggiornamenti

Per applicare un aggiornamento, è necessario sapere che ne è disponibile uno e quindi sapere come distribuirlo.

Per le build derivate AMIs quando ne AL2023 AMIs vengono rilasciate delle nuove, [EC2 Image Builder](#) può creare, applicare patch e testare automaticamente. AMIs Puoi attivare le tue pipeline di costruzione AMI o utilizzare AMIs [Ricevi notifiche sui nuovi aggiornamenti](#) la base.

Per applicare le patch in loco, puoi utilizzare strumenti come [AWS Systems Manager Patch Manager](#) per orchestrare l'applicazione degli aggiornamenti su un parco macchine.

Per quanto riguarda gli altri sistemi pubblici AMIs basati su AL2023, i relativi fornitori AMIs possono avere la propria pianificazione dei rilasci e i propri metodi di notifica. Quando utilizzi immagini derivate AMIs o contenitori, consulta la documentazione dell'editore su quando vengono rilasciati gli aggiornamenti.

Le modifiche apportate a ciascuna versione sono documentate nelle [note di AL2023 rilascio](#). Gli aggiornamenti di sicurezza sono pubblicati su [Amazon Linux Security Center \(ALAS\)](#).

Controlla gli aggiornamenti dei pacchetti disponibili nei AL2023 repository

Quando pubblichiamo una nuova versione dei AL2023 repository, tutte le versioni precedenti sono ancora disponibili. Per impostazione predefinita, il plug-in per la gestione delle versioni dei repository

si collega alla stessa versione utilizzata per sviluppare l'AMI. Se desideri controllare gli aggiornamenti dei pacchetti, procedi come segue.

1. Scopri le versioni dei repository disponibili eseguendo il comando riportato di seguito.

```
$ sudo dnf check-release-update
```

2. Seleziona una versione eseguendo il comando seguente.

```
$ sudo dnf upgrade --releasever=version
```

Questo comando avvia un aggiornamento utilizzando dnf dalla versione di rilascio attuale di Amazon Linux alla versione di rilascio specificata nella riga di comando. Un elenco degli aggiornamenti dei pacchetti è presentato da dnf. Prima di elaborare l'aggiornamento, devi confermarlo. Una volta completato l'aggiornamento, la nuova versione di rilascio diventa la versione di rilascio predefinita che dnf usa per tutte le attività future.

Per ulteriori informazioni, consulta [Gestisci gli aggiornamenti dei pacchetti e del sistema operativo in AL2023](#).

Aggiornamenti deterministici tramite sostituzione dell'istanza

La [Aggiornamenti deterministici tramite repository con versioni su AL2023](#) funzionalità di Amazon Linux 2023 rende la sostituzione delle istanze un modo semplice per implementare in modo deterministico e sicuro versioni aggiornate di AL2023. Gli aggiornamenti deterministici significano che, man mano che una nuova versione viene progressivamente implementata, se viene riscontrato un problema, è semplice tornare all'AMI precedente determinando la causa del problema.

Utilizzare la sostituzione delle istanze anziché applicare le patch in loco significa che gli aggiornamenti sono più deterministici e prevedibili, in quanto il lancio di nuove capacità può essere un percorso di codice ben collaudato con stati A e B chiari. Ciascuno degli stati precedente e successivo può essere ben testato in un sistema CI/CD prima dell'inizio della distribuzione.

Quando si eseguono le patch sul posto, ci sono molti stati intermedi tra prima e dopo l'applicazione degli aggiornamenti, il che è più difficile da testare per tutte le combinazioni di stati.

Una strategia di aggiornamento del sistema operativo che utilizza la sostituzione delle istanze con aggiornamenti deterministici si adatta bene ai modelli di distribuzione blu/green, a wave e basati su fasi.

Utilizzo degli aggiornamenti deterministici tramite repository con versioni

Argomenti

- [Uso di un sistema di aggiornamento deterministico](#)
- [Aggiornamento selettivo di un sistema con aggiornamento deterministico](#)
- [Uso dell'override persistente con aggiornamento deterministico](#)

Uso di un sistema di aggiornamento deterministico

Note

Il comportamento predefinito del gestore di pacchetti è cambiato da AL2

Gli aggiornamenti deterministici sono un modo efficace per garantire che tutte le modifiche agli ambienti di produzione possano essere completamente testate prima di un'implementazione su vasta scala. Ogni nuova AL2023 AMI è associata a una versione particolare di AL2023. Ciò fornisce un comportamento deterministico delle versioni dei pacchetti del sistema operativo installate all'avvio dell'AMI specifica. Gli aggiornamenti in loco possono riguardare una versione di rilascio specifica, garantendo un comportamento deterministico in tutta la flotta. Quando passate a versioni nuove AMIs o già aggiornate, potete testarle tutte nella vostra CI/CD pipeline, individuando eventuali problemi potenziali prima di distribuirle negli ambienti di produzione.

Puoi utilizzare strumenti come [AWS Systems Manager Patch Manager](#) per orchestrare l'applicazione degli aggiornamenti in un parco macchine. Per le build derivate AMIs quando ne AL2023 AMIs vengono rilasciate nuove, [EC2 Image Builder](#) può creare, applicare patch AMIs e testare automaticamente, oppure [Ricevi notifiche sui nuovi aggiornamenti](#) puoi sapere quando sono disponibili AMIs nuove basi o attivare le tue pipeline di creazione AMI.

Per informazioni su come limitare gli aggiornamenti a quelli di un particolare avviso, consulta [Applicazione degli aggiornamenti di sicurezza in loco](#)

Per applicare le patch in loco, è possibile utilizzare il gestore di pacchetti. `dnf` Quando esegui il comando `dnf upgrade`, il sistema verifica la presenza di aggiornamenti nel repository specificato dalla variabile `releasever`. Una versione valida `releasever` è una delle due *latest* o una versione con data, ad esempio. *2023.11.20260427*

È possibile modificare il valore di `releasever` utilizzando uno dei metodi descritti di seguito. Questi metodi sono elencati con priorità di sistema decrescente. Ciò significa che il metodo 1 sostituisce i metodi 2 e 3, e il metodo 2 sostituisce il metodo 3.

1. Il valore nel flag della riga di comando, `--releasever=latest`, se utilizzato.
2. Il valore specificato nel file della variabile di override, `/etc/dnf/vars/releasever`, se impostato.
3. La versione attualmente installata del pacchetto `system-release`.

Nell'esempio seguente, la versione è: **2023.0.20230210**

```
$ rpm -q system-release
system-release-2023.0.20230210-0.amzn2023.noarch
```

In un sistema appena installato, la variabile di override non è presente. Non sono disponibili aggiornamenti perché il sistema è bloccato sulla versione installata di `system-release`.

```
$ cat /etc/dnf/vars/releasever
cat: /etc/dnf/vars/releasever: No such file or directory
```

```
$ sudo dnf upgrade
Last metadata expiration check: 0:00:02 ago on Wed 15 Feb 2023 06:14:12 PM UTC.
Dependencies resolved.
Nothing to do.
Complete!
```

Puoi ottenere pacchetti di una versione specifica utilizzando il flag `releasever` per fornire la versione desiderata.

```
$ rpm -q system-release
system-release-2023.0.20230222-0.amzn2023.noarch
```

```
$ sudo dnf upgrade --releasever=2023.0.20230329
Amazon Linux 2023 repository                26 MB/s | 12 MB      00:00
Dependencies resolved.
=====
Package                Arch    Version                               Repository    Size
=====
```

```

Installing:
 kernel                aarch64 6.1.21-1.45.amzn2023      amazonlinux 26 M
Upgrading:
 amazon-linux-repo-s3  noarch  2023.0.20230329-0.amzn2023      amazonlinux 18 k
 ca-certificates      noarch  2023.2.60-1.0.amzn2023.0.1     amazonlinux 828 k
 cloud-init           noarch  22.2.2-1.amzn2023.1.7          amazonlinux 1.1 M

... [ list edited for clarity ]

system-release        noarch  2023.0.20230329-0.amzn2023      amazonlinux 29 k

... [ list edited for clarity ]

vim-data              noarch  2:9.0.1403-1.amzn2023.0.1      amazonlinux 25 k
vim-minimal           aarch64 2:9.0.1403-1.amzn2023.0.1      amazonlinux 753 k

Transaction Summary
=====
Install    1 Package
Upgrade   42 Packages

Total download size: 56 M

```

Poiché l'opzione `--releasever` sostituisce sia `system-release` che `/etc/dnf/vars/releasever`, il risultato di questo aggiornamento è il seguente:

1. L'aggiornamento sostituisce tutti i pacchetti installati che sono stati modificati tra la versione precedente e quella nuova.
2. L'aggiornamento blocca il sistema nel repository per la nuova versione di `system-release`.

Specificando sempre a quale versione `releasever` (ad esempio, AL2023 versione) eseguire l'aggiornamento, si ottiene una serie deterministica di modifiche in tutta la flotta. Hai lanciato la versione *A*, aggiornata a *B* e poi aggiornata a *C*

Aggiornamento selettivo di un sistema con aggiornamento deterministico

Note

Si consiglia di installare tutti gli aggiornamenti di una nuova versione anziché selezionare aggiornamenti specifici. L'applicazione solo di una parte di un aggiornamento al sistema

operativo dovrebbe costituire un'eccezione alla prassi standard di eseguire l'intero aggiornamento.

Potresti voler installare pacchetti selezionati da un rilascio recente, lasciando al contempo il sistema bloccato sulla versione di rilascio originale.

Puoi utilizzare `dnf check-update` per identificare i pacchetti di cui vuoi eseguire l'aggiornamento.

```
$ sudo dnf check-update --releasever=latest --security
Amazon Linux 2023 repository                13 MB/s | 10 MB    00:00
Last metadata expiration check: 0:00:02 ago on Wed 15 Feb 2023 02:52:21 AM UTC.

bind-libs.aarch64                32:9.16.27-1.amzn2023.0.1    amazonlinux
bind-license.noarch               32:9.16.27-1.amzn2023.0.1    amazonlinux
bind-utils.aarch64                32:9.16.27-1.amzn2023.0.1    amazonlinux
cryptsetup.aarch64                2.4.3-2.amzn2023.0.1        amazonlinux
cryptsetup-libs.aarch64           2.4.3-2.amzn2023.0.1        amazonlinux
curl-minimal.aarch64              7.85.0-1.amzn2023.0.1       amazonlinux
glibc.aarch64                     2.34-40.amzn2023.0.2        amazonlinux
glibc-all-langpacks.aarch64       2.34-40.amzn2023.0.2        amazonlinux
glibc-common.aarch64              2.34-40.amzn2023.0.2        amazonlinux
glibc-locale-source.aarch64       2.34-40.amzn2023.0.2        amazonlinux
gmp.aarch64                        1:6.2.1-2.amzn2023.0.1      amazonlinux
gnupg2-minimal.aarch64            2.3.7-1.amzn2023.0.2        amazonlinux
gzip.aarch64                       1.10-5.amzn2023.0.1         amazonlinux
kernel.aarch64                     6.1.12-17.42.amzn2023       amazonlinux
kernel-tools.aarch64              6.1.12-17.42.amzn2023       amazonlinux
libarchive.aarch64                 3.5.3-2.amzn2023.0.1        amazonlinux
libcurl-minimal.aarch64           7.85.0-1.amzn2023.0.1       amazonlinux
libsepol.aarch64                   3.4-3.amzn2023.0.2          amazonlinux
libsolv.aarch64                    0.7.22-1.amzn2023.0.1       amazonlinux
libxml2.aarch64                    2.9.14-1.amzn2023.0.1       amazonlinux
logrotate.aarch64                  3.20.1-2.amzn2023.0.2       amazonlinux
lua-libs.aarch64                   5.4.4-3.amzn2023.0.1        amazonlinux
lz4-libs.aarch64                   1.9.4-1.amzn2023.0.1        amazonlinux
openssl.aarch64                    1:3.0.5-1.amzn2023.0.3      amazonlinux
openssl-libs.aarch64              1:3.0.5-1.amzn2023.0.3      amazonlinux
pcre2.aarch64                       10.40-1.amzn2023.0.1        amazonlinux
pcre2-syntax.noarch                10.40-1.amzn2023.0.1        amazonlinux
rsync.aarch64                       3.2.6-1.amzn2023.0.2        amazonlinux
vim-common.aarch64                 2:9.0.475-1.amzn2023.0.1    amazonlinux
vim-data.noarch                     2:9.0.475-1.amzn2023.0.1    amazonlinux
```

<code>vim-enhanced.aarch64</code>	<code>2:9.0.475-1.amzn2023.0.1</code>	<code>amazonlinux</code>
<code>vim-filesystem.noarch</code>	<code>2:9.0.475-1.amzn2023.0.1</code>	<code>amazonlinux</code>
<code>vim-minimal.aarch64</code>	<code>2:9.0.475-1.amzn2023.0.1</code>	<code>amazonlinux</code>
<code>xz.aarch64</code>	<code>5.2.5-9.amzn2023.0.1</code>	<code>amazonlinux</code>
<code>xz-libs.aarch64</code>	<code>5.2.5-9.amzn2023.0.1</code>	<code>amazonlinux</code>
<code>zlib.aarch64</code>	<code>1.2.11-32.amzn2023.0.3</code>	<code>amazonlinux</code>

Installa i pacchetti di cui vuoi eseguire l'aggiornamento. Usa `sudo dnf upgrade --releasever=latest` e i nomi dei pacchetti per assicurarti che il pacchetto `system-release` rimanga invariato.

```
$ sudo dnf upgrade --releasever=latest openssl openssl-libs
```

```
Last metadata expiration check: 0:01:28 ago on Wed 15 Feb 2023 02:52:21 AM UTC.
```

```
Dependencies resolved.
```

```
=====
Package           Arch           Version                Repository           Size
=====
Upgrading:
openssl           aarch64       1:3.0.5-1.amzn2023.0.3  amazonlinux         1.1 M
openssl-libs     aarch64       1:3.0.5-1.amzn2023.0.3  amazonlinux         2.1 M
```

```
Transaction Summary
```

```
=====
Upgrade 2 Packages
```

```
Total download size: 3.2 M
```

Note

Usando `sudo dnf upgrade --releasever=latest` vengono aggiornati tutti i pacchetti, incluso `system-release`. Quindi, la versione rimane bloccata sul nuovo `system-release` a meno che non imposti l'override persistente.

Uso dell'override persistente con aggiornamento deterministico

Note

Con gli aggiornamenti deterministici, puoi integrare le modifiche del sistema operativo nella tua CI/CD pipeline. La disabilitazione degli aggiornamenti deterministici elimina la possibilità di eseguire test prima della distribuzione.

Invece di aggiungere `--releasever=latest`, è possibile utilizzare l'override persistente per sbloccare il sistema impostando il valore della variabile su `latest`. Utilizzandola sempre `latest`, questa opzione ripristina il comportamento del modello di AL2 aggiornamento, in base AL2023 al quale ogni chiamata al gestore di pacchetti controllerà sempre la versione più recente e non è limitata a nessuna versione specifica del sistema operativo.

Warning

Sbloccando il gestore di pacchetti utilizzando una sovrascrittura persistente degli aggiornamenti deterministici, correte il rischio di scoprire ogni possibile incompatibilità tra l'applicazione e un aggiornamento del sistema operativo in produzione.

Sebbene le incompatibilità siano rare, con un aggiornamento del sistema operativo si integrano nuove modifiche al codice nell'ambiente, i test di integrazione possono impedire l'implementazione di modifiche al codice che hanno un impatto negativo sugli ambienti di produzione.

```
$ echo latest | sudo tee /etc/dnf/vars/releasever
latest
```

```
$ sudo dnf upgrade
```

```
Last metadata expiration check: 0:03:36 ago on Wed 15 Feb 2023 02:52:21 AM UTC.
Dependencies resolved.
```

```
=====
Package                Arch    Version                               Repository    Size
=====
Installing:
kernel                  aarch64 6.1.73-45.135.amzn2023              amazonlinux  24 M
Upgrading:
acl                     aarch64 2.3.1-2.amzn2023.0.1                amazonlinux  72 k
```

alternatives	aarch64	1.15-2.amzn2023.0.1	amazonlinux	36 k
amazon-ec2-net-utils	noarch	2.3.0-1.amzn2023.0.1	amazonlinux	16 k
at	aarch64	3.1.23-6.amzn2023.0.1	amazonlinux	60 k
attr	aarch64	2.5.1-3.amzn2023.0.1	amazonlinux	59 k
audit	aarch64	3.0.6-1.amzn2023.0.1	amazonlinux	249 k
audit-libs	aarch64	3.0.6-1.amzn2023.0.1	amazonlinux	116 k
aws-c-auth-libs	aarch64	0.6.5-6.amzn2023.0.2	amazonlinux	79 k
aws-c-cal-libs	aarch64	0.5.12-7.amzn2023.0.2	amazonlinux	34 k
aws-c-common-libs	aarch64	0.6.14-6.amzn2023.0.2	amazonlinux	119 k
aws-c-compression-libs	aarch64	0.2.14-5.amzn2023.0.2	amazonlinux	22 k
aws-c-event-stream-libs	aarch64	0.2.7-5.amzn2023.0.2	amazonlinux	47 k
aws-c-http-libs	aarch64	0.6.8-6.amzn2023.0.2	amazonlinux	147 k
aws-c-io-libs	aarch64	0.10.12-5.amzn2023.0.6	amazonlinux	109 k
aws-c-mqtt-libs	aarch64	0.7.8-7.amzn2023.0.2	amazonlinux	61 k
aws-c-s3-libs	aarch64	0.1.27-5.amzn2023.0.3	amazonlinux	54 k
aws-c-sdkutils-libs	aarch64	0.1.1-5.amzn2023.0.2	amazonlinux	26 k
aws-checksums-libs	aarch64	0.1.12-5.amzn2023.0.2	amazonlinux	50 k
awscli-2	noarch	2.7.8-1.amzn2023.0.4	amazonlinux	7.3 M
basesystem	noarch	11-11.amzn2023.0.1	amazonlinux	7.8 k
bash	aarch64	5.1.8-2.amzn2023.0.1	amazonlinux	1.6 M
bash-completion	noarch	1:2.11-2.amzn2023.0.1	amazonlinux	292 k
bc	aarch64	1.07.1-14.amzn2023.0.1	amazonlinux	120 k
bind-libs	aarch64	32:9.16.27-1.amzn2023.0.1	amazonlinux	1.2 M
bind-license	noarch	32:9.16.27-1.amzn2023.0.1	amazonlinux	14 k
bind-utils	aarch64	32:9.16.27-1.amzn2023.0.1	amazonlinux	206 k
binutils	aarch64	2.38-20.amzn2023.0.3	amazonlinux	4.6 M
boost-filesystem	aarch64	1.75.0-4.amzn2023.0.1	amazonlinux	55 k
boost-system	aarch64	1.75.0-4.amzn2023.0.1	amazonlinux	14 k
boost-thread	aarch64	1.75.0-4.amzn2023.0.1	amazonlinux	54 k
bzip2	aarch64	1.0.8-6.amzn2023.0.1	amazonlinux	53 k
bzip2-libs	aarch64	1.0.8-6.amzn2023.0.1	amazonlinux	44 k
c-ares	aarch64	1.17.2-1.amzn2023.0.1	amazonlinux	107 k
ca-certificates	noarch	2021.2.50-1.0.amzn2023.0.3	amazonlinux	343 k
checkpolicy	aarch64	3.4-3.amzn2023.0.1	amazonlinux	345 k
chkconfig	aarch64	1.15-2.amzn2023.0.1	amazonlinux	162 k
chrony	aarch64	4.2-7.amzn2023.0.4	amazonlinux	314 k
cloud-init	noarch	22.2.2-1.amzn2023.1.7	amazonlinux	1.1 M
cloud-utils-growpart	aarch64	0.31-8.amzn2023.0.2	amazonlinux	31 k
coreutils	aarch64	8.32-30.amzn2023.0.2	amazonlinux	1.1 M
coreutils-common	aarch64	8.32-30.amzn2023.0.2	amazonlinux	2.0 M
cpio	aarch64	2.13-10.amzn2023.0.1	amazonlinux	269 k
cracklib	aarch64	2.9.6-27.amzn2023.0.1	amazonlinux	83 k
cracklib-dicts	aarch64	2.9.6-27.amzn2023.0.1	amazonlinux	3.6 M
crontabs	noarch	1.11-24.20190603git.amzn2023.0.1		

```

amazonlinux 19 k
crypto-policies          noarch 20230128-1.gitdfb10ea.amzn2023.0.1
amazonlinux 61 k
crypto-policies-scripts noarch 20230128-1.gitdfb10ea.amzn2023.0.1
amazonlinux 81 k
...
Installing dependencies:
amazon-linux-repo-cdn  noarch 2023.0.20230210-0.amzn2023  amazonlinux 16 k
xxhash-libs          aarch64 0.8.0-3.amzn2023.0.1        amazonlinux 32 k
Installing weak dependencies:
amazon-chrony-config noarch 4.2-7.amzn2023.0.4          amazonlinux 14 k
gawk-all-langpacks   aarch64 5.1.0-3.amzn2023.0.1        amazonlinux 207 k

Transaction Summary
=====
Install    5 Packages
Upgrade   413 Packages

Total download size: 199 M

```

Note

Se hai usato la variabile di override `/etc/dnf/vars/releasever`, utilizza il comando seguente per ripristinare il comportamento di blocco predefinito cancellando il valore di override.

```
$ sudo rm /etc/dnf/vars/releasever
```

L'uso di un override persistente `latest` rispetto a una versione specifica è simile al comportamento predefinito di AL2. Esistono servizi che AMIs si basano sui AL2 quali disabilitano questo comportamento e si limitano a versioni di pacchetti specifiche, come quelle impostate di default. AL2023

Invece di disabilitare gli aggiornamenti deterministici, consigliamo di sostituire le istanze con altre avviate da una nuova AMI. Se la sostituzione delle istanze non è un'opzione, consigliamo di utilizzare strumenti come [AWS Systems Manager Patch Manager](#) per orchestrare l'applicazione degli aggiornamenti in un parco istanze. [EC2 Image Builder](#) può anche creare, applicare patch e testare automaticamente le AMIs proprie immagini AL2023 derivate dalle immagini di base. Puoi anche [Ricevi notifiche sui nuovi aggiornamenti](#) utilizzarlo per attivare le tue pipeline di costruzione AMI.

L'utilizzo `latest` in un ambiente di preproduzione e la successiva distribuzione in produzione `latest` non forniscono protezione da eventuali problemi tra un aggiornamento del sistema operativo e l'applicazione. Una nuova AL2023 versione può essere rilasciata in qualsiasi momento, pertanto tutti gli utilizzi `latest` in produzione comportano dei rischi.

Gestisci gli aggiornamenti dei pacchetti e del sistema operativo in AL2023

A differenza delle versioni precedenti di Amazon Linux, AL2023 AMIs sono bloccati su una versione specifica del repository Amazon Linux. Per applicare correzioni di sicurezza e bug a un' AL2023 istanza, aggiorna la DNF configurazione all'ultima versione disponibile. In alternativa, avvia un'istanza più recente. AL2023

In questa sezione viene descritto come gestire repository e pacchetti DNF su un'istanza in esecuzione. Viene descritto inoltre come configurare DNF da uno script di dati utente per abilitare il repository Amazon Linux più recente disponibile al momento dell'avvio. Per ulteriori informazioni, consulta la sezione relativa alle [informazioni di riferimento ai comandi DNF](#).

Si consiglia di applicare tutti gli aggiornamenti disponibili in una nuova AL2023 versione. Scegliere solo gli aggiornamenti di sicurezza o solo gli aggiornamenti specifici dovrebbe essere l'eccezione piuttosto che la regola. Per un elenco degli [Avvisi di sicurezza](#) elementi pertinenti a una particolare istanza, consulta [Elenco degli avvisi applicabili](#). Per informazioni sull'installazione dei soli aggiornamenti relativi a un [avviso](#) specifico, vedere [Applicazione degli aggiornamenti di sicurezza in loco](#).

Important

Se desideri segnalare una vulnerabilità o hai un problema di sicurezza relativo ai servizi AWS cloud o ai progetti open source, contatta AWS Security using the [Vulnerability Reporting page](#)

Argomenti

- [Verifica degli aggiornamenti dei pacchetti disponibili](#)
- [Applicazione degli aggiornamenti di sicurezza utilizzando DNF e le versioni del repository](#)
- [Riavvio automatico del servizio dopo gli aggiornamenti \(di sicurezza\)](#)
- [Quando è necessario il riavvio per applicare gli aggiornamenti di sicurezza?](#)

- [Avvio di un'istanza con la versione più recente del repository abilitata](#)
- [Ottenere informazioni di supporto per i pacchetti](#)
- [Verifica della disponibilità di versioni più recenti del repository con `dnf check-release-update`](#)
- [Aggiunta, abilitazione o disabilitazione di nuovi repository](#)
- [Aggiunta di repository con `cloud-init`](#)

Verifica degli aggiornamenti dei pacchetti disponibili

Puoi utilizzare il comando `dnf check-update` per verificare la presenza di eventuali aggiornamenti per il sistema. Pertanto AL2023, ti consigliamo di aggiungere l'opzione `--releasever=version-number` al comando.

Quando aggiungi questa opzione, DNF verifica anche la presenza di aggiornamenti per una versione successiva del repository. Ad esempio, dopo aver eseguito il comando `dnf check-update`, usa la versione più recente restituita come valore per `version-number`.

Se l'istanza viene aggiornata per utilizzare la versione più recente del repository, l'output include un elenco di tutti i pacchetti da aggiornare.

Note

Se non specifichi la versione di rilascio con il flag opzionale al comando `dnf check-update`, viene controllata solo la versione del repository attualmente configurata. Ciò significa che i pacchetti nella versione successiva del repository non vengono controllati.

Updates in a specific version

[In questo esempio vedremo quali aggiornamenti sono disponibili nella versione 2023.1.20230628 se abbiamo lanciato un contenitore della versione 2023.0.20230315.](#)

Note

[Questo esempio utilizza le versioni 2023.0.20230315 e 2023.1.20230628, che non sono l'ultima versione di See the Release Notes per le ultime versioni, che contengono gli ultimi aggiornamenti di sicurezza. AL2023 AL2023](#)

[In questo esempio inizieremo con un'immagine del contenitore per la versione 2023.0.20230315.](#)

Innanzitutto, recuperiamo l'immagine del contenitore dal registro dei contenitori. La `.0` parte finale indica la versione dell'immagine per una particolare versione; questa versione dell'immagine di solito è zero.

```
$ docker pull public.ecr.aws/amazonlinux/amazonlinux:2023.0.20230315.0
2023.0.20230315.0: Pulling from amazonlinux/amazonlinux
b76f3b09316a: Pull complete
Digest: sha256:94e7183b0739140dbd5b639fb7600f0a2299cec5df8780c26d9cb409da5315a9
Status: Downloaded newer image for public.ecr.aws/amazonlinux/
amazonlinux:2023.0.20230315.0
public.ecr.aws/amazonlinux/amazonlinux:2023.0.20230315.0
```

Ora possiamo generare una shell all'interno del contenitore, dalla quale controlleremo gli aggiornamenti.

```
$ docker run -it public.ecr.aws/amazonlinux/amazonlinux:2023.0.20230315.0
bash-5.2#
```

Il `dnf check-update` comando viene ora utilizzato per controllare gli aggiornamenti disponibili nella versione [2023.1.20230628](#).

Note

L'applicazione degli aggiornamenti dei pacchetti è un'operazione privilegiata. Sebbene l'elevazione dei privilegi in genere non sia richiesta quando si esegue in un contenitore, se si esegue in un ambiente non containerizzato come un'istanza Amazon EC2, è possibile verificare la presenza di aggiornamenti senza elevare i privilegi.

```
$ dnf check-update --releasever=2023.1.20230628
Amazon Linux 2023 repository                60 MB/s | 15 MB      00:00
Last metadata expiration check: 0:00:02 ago on Mon Jul 22 17:25:34 2024.

amazon-linux-repo-cdn.noarch                2023.1.20230628-0.amzn2023      amazonlinux
ca-certificates.noarch                     2023.2.60-1.0.amzn2023.0.2     amazonlinux
curl-minimal.x86_64                         8.0.1-1.amzn2023               amazonlinux
glib2.x86_64                                2.74.7-688.amzn2023.0.1       amazonlinux
```

<code>glibc.x86_64</code>	<code>2.34-52.amzn2023.0.3</code>	<code>amazonlinux</code>
<code>glibc-common.x86_64</code>	<code>2.34-52.amzn2023.0.3</code>	<code>amazonlinux</code>
<code>glibc-minimal-langpack.x86_64</code>	<code>2.34-52.amzn2023.0.3</code>	<code>amazonlinux</code>
<code>gnupg2-minimal.x86_64</code>	<code>2.3.7-1.amzn2023.0.4</code>	<code>amazonlinux</code>
<code>keyutils-libs.x86_64</code>	<code>1.6.3-1.amzn2023</code>	<code>amazonlinux</code>
<code>libcap.x86_64</code>	<code>2.48-2.amzn2023.0.3</code>	<code>amazonlinux</code>
<code>libcurl-minimal.x86_64</code>	<code>8.0.1-1.amzn2023</code>	<code>amazonlinux</code>
<code>libgcc.x86_64</code>	<code>11.3.1-4.amzn2023.0.3</code>	<code>amazonlinux</code>
<code>libgomp.x86_64</code>	<code>11.3.1-4.amzn2023.0.3</code>	<code>amazonlinux</code>
<code>libstdc++.x86_64</code>	<code>11.3.1-4.amzn2023.0.3</code>	<code>amazonlinux</code>
<code>libxml2.x86_64</code>	<code>2.10.4-1.amzn2023.0.1</code>	<code>amazonlinux</code>
<code>ncurses-base.noarch</code>	<code>6.2-4.20200222.amzn2023.0.4</code>	<code>amazonlinux</code>
<code>ncurses-libs.x86_64</code>	<code>6.2-4.20200222.amzn2023.0.4</code>	<code>amazonlinux</code>
<code>openssl-libs.x86_64</code>	<code>1:3.0.8-1.amzn2023.0.3</code>	<code>amazonlinux</code>
<code>python3-rpm.x86_64</code>	<code>4.16.1.3-12.amzn2023.0.6</code>	<code>amazonlinux</code>
<code>rpm.x86_64</code>	<code>4.16.1.3-12.amzn2023.0.6</code>	<code>amazonlinux</code>
<code>rpm-build-libs.x86_64</code>	<code>4.16.1.3-12.amzn2023.0.6</code>	<code>amazonlinux</code>
<code>rpm-libs.x86_64</code>	<code>4.16.1.3-12.amzn2023.0.6</code>	<code>amazonlinux</code>
<code>rpm-sign-libs.x86_64</code>	<code>4.16.1.3-12.amzn2023.0.6</code>	<code>amazonlinux</code>
<code>system-release.noarch</code>	<code>2023.1.20230628-0.amzn2023</code>	<code>amazonlinux</code>
<code>tzdata.noarch</code>	<code>2023c-1.amzn2023.0.1</code>	<code>amazonlinux</code>
<code>bash-5.2#</code>		

La versione del `system-release` pacchetto mostra la versione a cui un `dnf upgrade` comando verrebbe aggiornato, ovvero la versione [2023.1.20230628](#) richiesta nel comando. `dnf check-update --releasever=2023.1.20230628`

Updates in the latest version

[In questo esempio vedremo quali aggiornamenti sono disponibili nella versione di AL2023 if abbiamo lanciato un contenitore della latest versione 2023.4.20240319.](#) Al momento della stesura, la latest versione è [2023.5.20240708](#), quindi gli aggiornamenti elencati in questo esempio saranno aggiornati a quella versione.

Note

Questo esempio utilizza le versioni [2023.4.20240319](#) e [2023.5.20240708](#), quest'ultima è [la versione più recente al momento](#) della stesura. [Per ulteriori informazioni sulle versioni più recenti, consulta le note di rilascio. AL2023](#)

In questo esempio inizieremo con un'immagine del contenitore per la versione [2023.4.20240319](#).

Innanzitutto, recuperiamo l'immagine del contenitore dal registro dei contenitori. La `.1` parte finale indica la versione dell'immagine per una particolare versione. Sebbene la versione dell'immagine sia in genere zero, questo esempio utilizza una versione in cui la versione dell'immagine è una.

```
$ docker pull public.ecr.aws/amazonlinux/amazonlinux:2023.4.20240319.1
2023.4.20240319.1: Pulling from amazonlinux/amazonlinux
6de065fda9a2: Pull complete
Digest: sha256:b4838c4cc9211d966b6ea158dacc9eda7433a16ba94436508c2d9f01f7658b4e
Status: Downloaded newer image for public.ecr.aws/amazonlinux/
amazonlinux:2023.4.20240319.1
public.ecr.aws/amazonlinux/amazonlinux:2023.4.20240319.1
```

Ora possiamo generare una shell all'interno del contenitore, dalla quale verificheremo la presenza di aggiornamenti.

```
$ docker run -it public.ecr.aws/amazonlinux/amazonlinux:2023.4.20240319.1
bash-5.2#
```

Il `dnf check-update` comando viene ora utilizzato per verificare gli aggiornamenti disponibili nella `latest` versione, che al momento della stesura era [2023.5.20240708](#).

Note

L'applicazione degli aggiornamenti dei pacchetti è un'operazione privilegiata. Sebbene l'elevazione dei privilegi in genere non sia richiesta quando si esegue in un contenitore, se si esegue in un ambiente non containerizzato come un'istanza Amazon EC2, è possibile verificare la presenza di aggiornamenti senza elevare i privilegi.

```
$ dnf --releasever=latest check-update
Amazon Linux 2023 repository                78 MB/s | 25 MB    00:00
Last metadata expiration check: 0:00:04 ago on Mon Jul 22 17:39:13 2024.

amazon-linux-repo-cdn.noarch                2023.5.20240708-1.amzn2023                amazonlinux
curl-minimal.x86_64                        8.5.0-1.amzn2023.0.4                      amazonlinux
dnf.noarch                                  4.14.0-1.amzn2023.0.5                    amazonlinux
dnf-data.noarch                             4.14.0-1.amzn2023.0.5                    amazonlinux
expat.x86_64                                2.5.0-1.amzn2023.0.4                      amazonlinux
glibc.x86_64                                2.34-52.amzn2023.0.10                    amazonlinux
glibc-common.x86_64                        2.34-52.amzn2023.0.10                    amazonlinux
```

```

glibc-minimal-langpack.x86_64      2.34-52.amzn2023.0.10      amazonlinux
krb5-libs.x86_64                   1.21-3.amzn2023.0.4       amazonlinux
libblkid.x86_64                     2.37.4-1.amzn2023.0.4    amazonlinux
libcurl-minimal.x86_64              8.5.0-1.amzn2023.0.4     amazonlinux
libmount.x86_64                     2.37.4-1.amzn2023.0.4    amazonlinux
libnghttp2.x86_64                   1.59.0-3.amzn2023.0.1    amazonlinux
libsmartcols.x86_64                2.37.4-1.amzn2023.0.4    amazonlinux
libuuid.x86_64                      2.37.4-1.amzn2023.0.4    amazonlinux
openssl-libs.x86_64                1:3.0.8-1.amzn2023.0.12  amazonlinux
python3.x86_64                      3.9.16-1.amzn2023.0.8    amazonlinux
python3-dnf.noarch                  4.14.0-1.amzn2023.0.5    amazonlinux
python3-libs.x86_64                3.9.16-1.amzn2023.0.8    amazonlinux
system-release.noarch               2023.5.20240708-1.amzn2023 amazonlinux
yum.noarch                           4.14.0-1.amzn2023.0.5    amazonlinux
bash-5.2#

```

La versione del `system-release` pacchetto mostra la versione a cui un comando verrebbe aggiornato. `dnf upgrade`

Per questo comando, se sono disponibili pacchetti più recenti, il codice restituito è 100. Se non sono disponibili pacchetti più recenti, il codice restituito è 0. Inoltre, l'output elenca anche tutti i pacchetti da aggiornare.

Applicazione degli aggiornamenti di sicurezza utilizzando DNF e le versioni del repository

I nuovi aggiornamenti dei pacchetti e gli aggiornamenti di sicurezza sono disponibili solo per le nuove versioni del repository. Per i casi avviati da versioni AL2023 AMI precedenti, è necessario aggiornare la versione del repository prima di poter installare gli aggiornamenti di sicurezza. Il comando `dnf check-release-update` include un comando di aggiornamento di esempio che aggiorna tutti i pacchetti installati sul sistema alle versioni in un repository più recente.

Note

Se non specifichi la versione di rilascio con il flag opzionale al comando `dnf check-update`, viene controllata solo la versione del repository attualmente configurata. Ciò significa che qualsiasi aggiornamento ai pacchetti installati presenti in qualsiasi versione successiva del repository non viene applicato.

Questa sezione illustra il percorso di aggiornamento consigliato per applicare tutti gli aggiornamenti disponibili anziché scegliere singoli aggiornamenti o solo quelli contrassegnati come aggiornamenti di sicurezza. Applicando tutti gli aggiornamenti, le istanze esistenti vengono spostate nello stesso set di pacchetti utilizzato per l'avvio di un'AMI aggiornata. Questa coerenza riduce la variazione delle versioni dei pacchetti all'interno di una flotta. Per ulteriori informazioni sull'applicazione di aggiornamenti specifici, consulta [Applicazione degli aggiornamenti di sicurezza in loco](#).

Applying updates in a specific version

[In questo esempio applicheremo gli aggiornamenti disponibili nella versione 2023.1.20230628 se lanciamo un contenitore della versione 2023.0.20230315.](#)

Note

[Questo esempio utilizza le versioni 2023.0.20230315 e 2023.1.20230628, che non sono le versioni più recenti di See the Release Notes per le ultime versioni, che contengono gli ultimi aggiornamenti di sicurezza. AL2023 AL2023](#)

[In questo esempio inizieremo con un'immagine del contenitore per la versione 2023.0.20230315.](#)

Innanzitutto, recuperiamo l'immagine del contenitore dal registro dei contenitori. La `.0` parte finale indica la versione dell'immagine per una particolare versione; questa versione dell'immagine di solito è zero.

```
$ docker pull public.ecr.aws/amazonlinux/amazonlinux:2023.0.20230315.0
2023.0.20230315.0: Pulling from amazonlinux/amazonlinux
b76f3b09316a: Pull complete
Digest: sha256:94e7183b0739140dbd5b639fb7600f0a2299cec5df8780c26d9cb409da5315a9
Status: Downloaded newer image for public.ecr.aws/amazonlinux/
amazonlinux:2023.0.20230315.0
public.ecr.aws/amazonlinux/amazonlinux:2023.0.20230315.0
```

Ora possiamo generare una shell all'interno del contenitore, dalla quale applicheremo gli aggiornamenti.

```
$ docker run -it public.ecr.aws/amazonlinux/amazonlinux:2023.0.20230315.0
bash-5.2#
```

Il `dnf upgrade` comando viene ora utilizzato per applicare tutti gli aggiornamenti presenti nella versione [2023.1.20230628](#).

Note

L'applicazione degli aggiornamenti dei pacchetti è un'operazione privilegiata. Sebbene l'elevazione dei privilegi non sia in genere richiesta quando si esegue in un contenitore, se si esegue in un ambiente non containerizzato come un'istanza Amazon EC2, sarà necessario eseguire il comando come utente. `dnf upgrade root` Questa operazione può essere eseguita utilizzando i comandi `or. sudo su`

```
$ dnf upgrade --releasever=2023.1.20230628
```

```
Amazon Linux 2023 repository                38 MB/s | 15 MB    00:00
```

```
Last metadata expiration check: 0:00:02 ago on Mon Jul 22 17:49:08 2024.
```

```
Dependencies resolved.
```

```
=====
```

Package	Arch	Version	Repository	Size
Upgrading:				
amazon-linux-repo-cdn	noarch	2023.1.20230628-0.amzn2023	amazonlinux	18 k
ca-certificates	noarch	2023.2.60-1.0.amzn2023.0.2	amazonlinux	829 k
curl-minimal	x86_64	8.0.1-1.amzn2023	amazonlinux	150 k
glib2	x86_64	2.74.7-688.amzn2023.0.1	amazonlinux	2.7 M
glibc	x86_64	2.34-52.amzn2023.0.3	amazonlinux	1.9 M
glibc-common	x86_64	2.34-52.amzn2023.0.3	amazonlinux	307 k
glibc-minimal-langpack	x86_64	2.34-52.amzn2023.0.3	amazonlinux	35 k
gnupg2-minimal	x86_64	2.3.7-1.amzn2023.0.4	amazonlinux	421 k
keyutils-libs	x86_64	1.6.3-1.amzn2023	amazonlinux	33 k
libcap	x86_64	2.48-2.amzn2023.0.3	amazonlinux	67 k
libcurl-minimal	x86_64	8.0.1-1.amzn2023	amazonlinux	249 k
libgcc	x86_64	11.3.1-4.amzn2023.0.3	amazonlinux	105 k
libgomp	x86_64	11.3.1-4.amzn2023.0.3	amazonlinux	280 k
libstdc++	x86_64	11.3.1-4.amzn2023.0.3	amazonlinux	744 k
libxml2	x86_64	2.10.4-1.amzn2023.0.1	amazonlinux	706 k
ncurses-base	noarch	6.2-4.20200222.amzn2023.0.4	amazonlinux	60 k
ncurses-libs	x86_64	6.2-4.20200222.amzn2023.0.4	amazonlinux	328 k
openssl-libs	x86_64	1:3.0.8-1.amzn2023.0.3	amazonlinux	2.2 M
python3-rpm	x86_64	4.16.1.3-12.amzn2023.0.6	amazonlinux	88 k
rpm	x86_64	4.16.1.3-12.amzn2023.0.6	amazonlinux	486 k
rpm-build-libs	x86_64	4.16.1.3-12.amzn2023.0.6	amazonlinux	90 k
rpm-libs	x86_64	4.16.1.3-12.amzn2023.0.6	amazonlinux	309 k

```

rpm-sign-libs          x86_64 4.16.1.3-12.amzn2023.0.6   amazonlinux 21 k
system-release        noarch 2023.1.20230628-0.amzn2023   amazonlinux 29 k
tzdata                noarch 2023c-1.amzn2023.0.1         amazonlinux 433 k

```

Transaction Summary

```
=====
```

```
Upgrade 25 Packages
```

```
Total download size: 12 M
```

```
Is this ok [y/N]:
```

La versione del `system-release` pacchetto mostra la versione a cui un `dnf upgrade` comando verrebbe aggiornato, ovvero la versione [2023.1.20230628 richiesta](#) nel comando. `dnf upgrade --releasever=2023.1.20230628`

Per impostazione predefinita, ti `dnf` chiederà di confermare che desideri applicare gli aggiornamenti. È possibile ignorare questa richiesta utilizzando il `-y` flag `dnf`. Per questo esempio, il `dnf upgrade -y --releasever=2023.1.20230628` comando non chiederà conferma prima di applicare gli aggiornamenti. Ciò è utile negli script o in altri ambienti di automazione.

Dopo aver confermato di voler applicare gli aggiornamenti, li `dnf` applica.

```
Is this ok [y/N]:y
```

```
  Downloading Packages:
```

```

(1/25): libcap-2.48-2.amzn2023.0.3.x86_64.rpm    1.5 MB/s | 67 kB    00:00
(2/25): python3-rpm-4.16.1.3-12.amzn2023.0.6.x86 2.1 MB/s | 88 kB    00:00
(3/25): libcurl-minimal-8.0.1-1.amzn2023.x86_64. 2.6 MB/s | 249 kB   00:00
(4/25): glib2-2.74.7-688.amzn2023.0.1.x86_64.rpm 26 MB/s | 2.7 MB   00:00
(5/25): glibc-minimal-langpack-2.34-52.amzn2023. 1.3 MB/s | 35 kB    00:00
(6/25): rpm-build-libs-4.16.1.3-12.amzn2023.0.6. 2.8 MB/s | 90 kB    00:00
(7/25): rpm-libs-4.16.1.3-12.amzn2023.0.6.x86_64 6.6 MB/s | 309 kB   00:00
(8/25): libgcc-11.3.1-4.amzn2023.0.3.x86_64.rpm 3.9 MB/s | 105 kB   00:00
(9/25): glibc-common-2.34-52.amzn2023.0.3.x86_64 11 MB/s | 307 kB    00:00
(10/25): glibc-2.34-52.amzn2023.0.3.x86_64.rpm   31 MB/s | 1.9 MB    00:00
(11/25): rpm-sign-libs-4.16.1.3-12.amzn2023.0.6. 877 kB/s | 21 kB    00:00
(12/25): gnupg2-minimal-2.3.7-1.amzn2023.0.4.x86 15 MB/s | 421 kB    00:00
(13/25): openssl-libs-3.0.8-1.amzn2023.0.3.x86_6 35 MB/s | 2.2 MB    00:00
(14/25): libxml2-2.10.4-1.amzn2023.0.1.x86_64.rp 14 MB/s | 706 kB    00:00
(15/25): curl-minimal-8.0.1-1.amzn2023.x86_64.rp 4.2 MB/s | 150 kB   00:00
(16/25): rpm-4.16.1.3-12.amzn2023.0.6.x86_64.rpm 11 MB/s | 486 kB    00:00
(17/25): libgomp-11.3.1-4.amzn2023.0.3.x86_64.rp 7.0 MB/s | 280 kB   00:00

```

```
(18/25): libstdc++-11.3.1-4.amzn2023.0.3.x86_64. 14 MB/s | 744 kB 00:00
(19/25): keyutils-libs-1.6.3-1.amzn2023.x86_64.r 1.6 MB/s | 33 kB 00:00
(20/25): ncurses-libs-6.2-4.20200222.amzn2023.0. 10 MB/s | 328 kB 00:00
(21/25): tzdata-2023c-1.amzn2023.0.1.noarch.rpm 11 MB/s | 433 kB 00:00
(22/25): amazon-linux-repo-cdn-2023.1.20230628-0 781 kB/s | 18 kB 00:00
(23/25): ca-certificates-2023.2.60-1.0.amzn2023. 16 MB/s | 829 kB 00:00
(24/25): system-release-2023.1.20230628-0.amzn20 1.5 MB/s | 29 kB 00:00
(25/25): ncurses-base-6.2-4.20200222.amzn2023.0. 3.1 MB/s | 60 kB 00:00
```

```
-----
Total                               28 MB/s | 12 MB 00:00
```

Running transaction check

Transaction check succeeded.

Running transaction test

Transaction test succeeded.

Running transaction

```
Preparing      : 1/1
Upgrading      : libgcc-11.3.1-4.amzn2023.0.3.x86_64 1/50
Running scriptlet: libgcc-11.3.1-4.amzn2023.0.3.x86_64 1/50
Upgrading      : system-release-2023.1.20230628-0.amzn2023.noarch 2/50
Upgrading      : amazon-linux-repo-cdn-2023.1.20230628-0.amzn2023.no 3/50
Upgrading      : ncurses-base-6.2-4.20200222.amzn2023.0.4.noarch 4/50
Upgrading      : tzdata-2023c-1.amzn2023.0.1.noarch 5/50
Upgrading      : glibc-common-2.34-52.amzn2023.0.3.x86_64 6/50
Running scriptlet: glibc-2.34-52.amzn2023.0.3.x86_64 7/50
Upgrading      : glibc-2.34-52.amzn2023.0.3.x86_64 7/50
Running scriptlet: glibc-2.34-52.amzn2023.0.3.x86_64 7/50
Upgrading      : glibc-minimal-langpack-2.34-52.amzn2023.0.3.x86_64 8/50
Upgrading      : libcap-2.48-2.amzn2023.0.3.x86_64 9/50
Upgrading      : gnupg2-minimal-2.3.7-1.amzn2023.0.4.x86_64 10/50
Upgrading      : libgomp-11.3.1-4.amzn2023.0.3.x86_64 11/50
Running scriptlet: ca-certificates-2023.2.60-1.0.amzn2023.0.2.noarch 12/50
Upgrading      : ca-certificates-2023.2.60-1.0.amzn2023.0.2.noarch 12/50
Running scriptlet: ca-certificates-2023.2.60-1.0.amzn2023.0.2.noarch 12/50
Upgrading      : openssl-libs-1:3.0.8-1.amzn2023.0.3.x86_64 13/50
Upgrading      : libcurl-minimal-8.0.1-1.amzn2023.x86_64 14/50
Upgrading      : curl-minimal-8.0.1-1.amzn2023.x86_64 15/50
Upgrading      : rpm-libs-4.16.1.3-12.amzn2023.0.6.x86_64 16/50
Upgrading      : rpm-4.16.1.3-12.amzn2023.0.6.x86_64 17/50
Upgrading      : rpm-build-libs-4.16.1.3-12.amzn2023.0.6.x86_64 18/50
Upgrading      : rpm-sign-libs-4.16.1.3-12.amzn2023.0.6.x86_64 19/50
Upgrading      : python3-rpm-4.16.1.3-12.amzn2023.0.6.x86_64 20/50
Upgrading      : glib2-2.74.7-688.amzn2023.0.1.x86_64 21/50
Upgrading      : libxml2-2.10.4-1.amzn2023.0.1.x86_64 22/50
Upgrading      : libstdc++-11.3.1-4.amzn2023.0.3.x86_64 23/50
```

Upgrading	: keyutils-libs-1.6.3-1.amzn2023.x86_64	24/50
Upgrading	: ncurses-libs-6.2-4.20200222.amzn2023.0.4.x86_64	25/50
Cleanup	: glib2-2.73.2-680.amzn2023.0.3.x86_64	26/50
Cleanup	: libstdc++-11.3.1-4.amzn2023.0.2.x86_64	27/50
Cleanup	: libxml2-2.10.3-2.amzn2023.0.1.x86_64	28/50
Cleanup	: python3-rpm-4.16.1.3-12.amzn2023.0.5.x86_64	29/50
Cleanup	: rpm-build-libs-4.16.1.3-12.amzn2023.0.5.x86_64	30/50
Cleanup	: rpm-sign-libs-4.16.1.3-12.amzn2023.0.5.x86_64	31/50
Cleanup	: rpm-libs-4.16.1.3-12.amzn2023.0.5.x86_64	32/50
Cleanup	: libcap-2.48-2.amzn2023.0.2.x86_64	33/50
Cleanup	: gnupg2-minimal-2.3.7-1.amzn2023.0.3.x86_64	34/50
Cleanup	: ncurses-libs-6.2-4.20200222.amzn2023.0.3.x86_64	35/50
Cleanup	: libgomp-11.3.1-4.amzn2023.0.2.x86_64	36/50
Cleanup	: rpm-4.16.1.3-12.amzn2023.0.5.x86_64	37/50
Cleanup	: curl-minimal-7.88.1-1.amzn2023.0.1.x86_64	38/50
Cleanup	: libcurl-minimal-7.88.1-1.amzn2023.0.1.x86_64	39/50
Cleanup	: openssl-libs-1:3.0.8-1.amzn2023.0.1.x86_64	40/50
Cleanup	: keyutils-libs-1.6.1-2.amzn2023.0.2.x86_64	41/50
Cleanup	: amazon-linux-repo-cdn-2023.0.20230315-1.amzn2023.no	42/50
Cleanup	: system-release-2023.0.20230315-1.amzn2023.noarch	43/50
Cleanup	: ca-certificates-2023.2.60-1.0.amzn2023.0.1.noarch	44/50
Cleanup	: ncurses-base-6.2-4.20200222.amzn2023.0.3.noarch	45/50
Cleanup	: glibc-minimal-langpack-2.34-52.amzn2023.0.2.x86_64	46/50
Cleanup	: glibc-2.34-52.amzn2023.0.2.x86_64	47/50
Cleanup	: glibc-common-2.34-52.amzn2023.0.2.x86_64	48/50
Cleanup	: tzdata-2022g-1.amzn2023.0.1.noarch	49/50
Cleanup	: libgcc-11.3.1-4.amzn2023.0.2.x86_64	50/50
Running scriptlet:	libgcc-11.3.1-4.amzn2023.0.2.x86_64	50/50
Running scriptlet:	ca-certificates-2023.2.60-1.0.amzn2023.0.2.noarch	50/50
Running scriptlet:	rpm-4.16.1.3-12.amzn2023.0.6.x86_64	50/50
Running scriptlet:	libgcc-11.3.1-4.amzn2023.0.2.x86_64	50/50
Verifying	: libcurl-minimal-8.0.1-1.amzn2023.x86_64	1/50
Verifying	: libcurl-minimal-7.88.1-1.amzn2023.0.1.x86_64	2/50
Verifying	: libcap-2.48-2.amzn2023.0.3.x86_64	3/50
Verifying	: libcap-2.48-2.amzn2023.0.2.x86_64	4/50
Verifying	: glib2-2.74.7-688.amzn2023.0.1.x86_64	5/50
Verifying	: glib2-2.73.2-680.amzn2023.0.3.x86_64	6/50
Verifying	: python3-rpm-4.16.1.3-12.amzn2023.0.6.x86_64	7/50
Verifying	: python3-rpm-4.16.1.3-12.amzn2023.0.5.x86_64	8/50
Verifying	: glibc-minimal-langpack-2.34-52.amzn2023.0.3.x86_64	9/50
Verifying	: glibc-minimal-langpack-2.34-52.amzn2023.0.2.x86_64	10/50
Verifying	: rpm-libs-4.16.1.3-12.amzn2023.0.6.x86_64	11/50
Verifying	: rpm-libs-4.16.1.3-12.amzn2023.0.5.x86_64	12/50
Verifying	: rpm-build-libs-4.16.1.3-12.amzn2023.0.6.x86_64	13/50

```

Verifying      : rpm-build-libs-4.16.1.3-12.amzn2023.0.5.x86_64      14/50
Verifying      : glibc-2.34-52.amzn2023.0.3.x86_64                 15/50
Verifying      : glibc-2.34-52.amzn2023.0.2.x86_64                 16/50
Verifying      : libgcc-11.3.1-4.amzn2023.0.3.x86_64                17/50
Verifying      : libgcc-11.3.1-4.amzn2023.0.2.x86_64                18/50
Verifying      : glibc-common-2.34-52.amzn2023.0.3.x86_64           19/50
Verifying      : glibc-common-2.34-52.amzn2023.0.2.x86_64           20/50
Verifying      : rpm-sign-libs-4.16.1.3-12.amzn2023.0.6.x86_64      21/50
Verifying      : rpm-sign-libs-4.16.1.3-12.amzn2023.0.5.x86_64      22/50
Verifying      : openssl-libs-1:3.0.8-1.amzn2023.0.3.x86_64         23/50
Verifying      : openssl-libs-1:3.0.8-1.amzn2023.0.1.x86_64         24/50
Verifying      : gnupg2-minimal-2.3.7-1.amzn2023.0.4.x86_64         25/50
Verifying      : gnupg2-minimal-2.3.7-1.amzn2023.0.3.x86_64         26/50
Verifying      : libxml2-2.10.4-1.amzn2023.0.1.x86_64                27/50
Verifying      : libxml2-2.10.3-2.amzn2023.0.1.x86_64                28/50
Verifying      : curl-minimal-8.0.1-1.amzn2023.x86_64                29/50
Verifying      : curl-minimal-7.88.1-1.amzn2023.0.1.x86_64          30/50
Verifying      : rpm-4.16.1.3-12.amzn2023.0.6.x86_64                31/50
Verifying      : rpm-4.16.1.3-12.amzn2023.0.5.x86_64                32/50
Verifying      : libstdc++-11.3.1-4.amzn2023.0.3.x86_64              33/50
Verifying      : libstdc++-11.3.1-4.amzn2023.0.2.x86_64              34/50
Verifying      : libgomp-11.3.1-4.amzn2023.0.3.x86_64                35/50
Verifying      : libgomp-11.3.1-4.amzn2023.0.2.x86_64                36/50
Verifying      : keyutils-libs-1.6.3-1.amzn2023.x86_64                37/50
Verifying      : keyutils-libs-1.6.1-2.amzn2023.0.2.x86_64           38/50
Verifying      : ncurses-libs-6.2-4.20200222.amzn2023.0.4.x86_64     39/50
Verifying      : ncurses-libs-6.2-4.20200222.amzn2023.0.3.x86_64     40/50
Verifying      : ca-certificates-2023.2.60-1.0.amzn2023.0.2.noarch   41/50
Verifying      : ca-certificates-2023.2.60-1.0.amzn2023.0.1.noarch   42/50
Verifying      : tzdata-2023c-1.amzn2023.0.1.noarch                  43/50
Verifying      : tzdata-2022g-1.amzn2023.0.1.noarch                  44/50
Verifying      : amazon-linux-repo-cdn-2023.1.20230628-0.amzn2023.no  45/50
Verifying      : amazon-linux-repo-cdn-2023.0.20230315-1.amzn2023.no  46/50
Verifying      : system-release-2023.1.20230628-0.amzn2023.noarch    47/50
Verifying      : system-release-2023.0.20230315-1.amzn2023.noarch    48/50
Verifying      : ncurses-base-6.2-4.20200222.amzn2023.0.4.noarch     49/50
Verifying      : ncurses-base-6.2-4.20200222.amzn2023.0.3.noarch     50/50

```

Upgraded:

```

amazon-linux-repo-cdn-2023.1.20230628-0.amzn2023.noarch
ca-certificates-2023.2.60-1.0.amzn2023.0.2.noarch
curl-minimal-8.0.1-1.amzn2023.x86_64
glib2-2.74.7-688.amzn2023.0.1.x86_64
glibc-2.34-52.amzn2023.0.3.x86_64

```

```
glibc-common-2.34-52.amzn2023.0.3.x86_64
glibc-minimal-langpack-2.34-52.amzn2023.0.3.x86_64
gnupg2-minimal-2.3.7-1.amzn2023.0.4.x86_64
keyutils-libs-1.6.3-1.amzn2023.x86_64
libcap-2.48-2.amzn2023.0.3.x86_64
libcurl-minimal-8.0.1-1.amzn2023.x86_64
libgcc-11.3.1-4.amzn2023.0.3.x86_64
libgomp-11.3.1-4.amzn2023.0.3.x86_64
libstdc++-11.3.1-4.amzn2023.0.3.x86_64
libxml2-2.10.4-1.amzn2023.0.1.x86_64
ncurses-base-6.2-4.20200222.amzn2023.0.4.noarch
ncurses-libs-6.2-4.20200222.amzn2023.0.4.x86_64
openssl-libs-1:3.0.8-1.amzn2023.0.3.x86_64
python3-rpm-4.16.1.3-12.amzn2023.0.6.x86_64
rpm-4.16.1.3-12.amzn2023.0.6.x86_64
rpm-build-libs-4.16.1.3-12.amzn2023.0.6.x86_64
rpm-libs-4.16.1.3-12.amzn2023.0.6.x86_64
rpm-sign-libs-4.16.1.3-12.amzn2023.0.6.x86_64
system-release-2023.1.20230628-0.amzn2023.noarch
tzdata-2023c-1.amzn2023.0.1.noarch
```

Complete!

```
bash-5.2#
```

Updates in the latest version

In questo esempio applicheremo gli aggiornamenti disponibili nella latest versione di AL2023 if we launched a container della release [2023.4.20240319](#). Al momento della stesura, la latest versione è [2023.5.20240708](#), quindi gli aggiornamenti elencati in questo esempio saranno aggiornati a quella versione.

Note

Questo esempio utilizza le versioni [2023.4.20240319](#) e [2023.5.20240708](#), quest'ultima è [la versione più recente al momento](#) della stesura. [Per ulteriori informazioni sulle versioni più recenti, consulta le note di rilascio. AL2023](#)

In questo esempio inizieremo con un'immagine del contenitore per la versione [2023.4.20240319](#).

Innanzitutto, recuperiamo l'immagine del contenitore dal registro dei contenitori. La `.1` parte finale indica la versione dell'immagine per una particolare versione. Sebbene la versione dell'immagine sia in genere zero, questo esempio utilizza una versione in cui la versione dell'immagine è una.

```
$ docker pull public.ecr.aws/amazonlinux/amazonlinux:2023.4.20240319.1
2023.4.20240319.1: Pulling from amazonlinux/amazonlinux
6de065fda9a2: Pull complete
Digest: sha256:b4838c4cc9211d966b6ea158dacc9eda7433a16ba94436508c2d9f01f7658b4e
Status: Downloaded newer image for public.ecr.aws/amazonlinux/
amazonlinux:2023.4.20240319.1
public.ecr.aws/amazonlinux/amazonlinux:2023.4.20240319.1
```

Ora possiamo generare una shell all'interno del contenitore, dalla quale applicheremo gli aggiornamenti.

```
$ docker run -it public.ecr.aws/amazonlinux/amazonlinux:2023.4.20240319.1
bash-5.2#
```

Il `dnf upgrade` comando viene ora utilizzato per applicare gli aggiornamenti disponibili nella `latest` versione, che al momento della stesura era [2023.5.20240708](#).

Note

L'applicazione degli aggiornamenti dei pacchetti è un'operazione privilegiata. Sebbene l'elevazione dei privilegi non sia in genere richiesta quando si esegue in un contenitore, se si esegue in un ambiente non containerizzato come un'istanza Amazon EC2, sarà necessario eseguire il comando come utente. `dnf upgrade root` Questa operazione può essere eseguita utilizzando i comandi `or. sudo su`

Per impostazione predefinita, ti `dnf` verrà chiesto di confermare che desideri applicare gli aggiornamenti. In questo esempio, stiamo ignorando questa richiesta utilizzando il `-y` flag to. `dnf`

```
$ dnf -y --releasever=latest update
Amazon Linux 2023 repository                75 MB/s | 25 MB    00:00
Last metadata expiration check: 0:00:04 ago on Mon Jul 22 18:00:10 2024.
Dependencies resolved.
=====
Package                Arch    Version                                Repository    Size
=====
```

Upgrading:

amazon-linux-repo-cdn	noarch	2023.5.20240708-1.amzn2023	amazonlinux	17 k
curl-minimal	x86_64	8.5.0-1.amzn2023.0.4	amazonlinux	160 k
dnf	noarch	4.14.0-1.amzn2023.0.5	amazonlinux	460 k
dnf-data	noarch	4.14.0-1.amzn2023.0.5	amazonlinux	34 k
expat	x86_64	2.5.0-1.amzn2023.0.4	amazonlinux	117 k
glibc	x86_64	2.34-52.amzn2023.0.10	amazonlinux	1.9 M
glibc-common	x86_64	2.34-52.amzn2023.0.10	amazonlinux	295 k
glibc-minimal-langpack	x86_64	2.34-52.amzn2023.0.10	amazonlinux	23 k
krb5-libs	x86_64	1.21-3.amzn2023.0.4	amazonlinux	758 k
libblkid	x86_64	2.37.4-1.amzn2023.0.4	amazonlinux	105 k
libcurl-minimal	x86_64	8.5.0-1.amzn2023.0.4	amazonlinux	275 k
libmount	x86_64	2.37.4-1.amzn2023.0.4	amazonlinux	132 k
libnghttp2	x86_64	1.59.0-3.amzn2023.0.1	amazonlinux	79 k
libsmartcols	x86_64	2.37.4-1.amzn2023.0.4	amazonlinux	62 k
libuuid	x86_64	2.37.4-1.amzn2023.0.4	amazonlinux	26 k
openssl-libs	x86_64	1:3.0.8-1.amzn2023.0.12	amazonlinux	2.2 M
python3	x86_64	3.9.16-1.amzn2023.0.8	amazonlinux	27 k
python3-dnf	noarch	4.14.0-1.amzn2023.0.5	amazonlinux	409 k
python3-libs	x86_64	3.9.16-1.amzn2023.0.8	amazonlinux	7.3 M
system-release	noarch	2023.5.20240708-1.amzn2023	amazonlinux	28 k
yum	noarch	4.14.0-1.amzn2023.0.5	amazonlinux	32 k

Transaction Summary

```
=====
Upgrade 21 Packages
```

```
Total download size: 14 M
```

```
Downloading Packages:
```

```
(1/21): amazon-linux-repo-cdn-2023.5.20240708-1. 345 kB/s | 17 kB    00:00
(2/21): dnf-4.14.0-1.amzn2023.0.5.noarch.rpm    6.8 MB/s | 460 kB    00:00
(3/21): dnf-data-4.14.0-1.amzn2023.0.5.noarch.rp 1.6 MB/s | 34 kB    00:00
(4/21): expat-2.5.0-1.amzn2023.0.4.x86_64.rpm  4.6 MB/s | 117 kB    00:00
(5/21): glibc-2.34-52.amzn2023.0.10.x86_64.rpm 38 MB/s | 1.9 MB    00:00
(6/21): glibc-common-2.34-52.amzn2023.0.10.x86_6 8.8 MB/s | 295 kB    00:00
(7/21): glibc-minimal-langpack-2.34-52.amzn2023. 1.7 MB/s | 23 kB    00:00
(8/21): curl-minimal-8.5.0-1.amzn2023.0.4.x86_64 998 kB/s | 160 kB    00:00
(9/21): libblkid-2.37.4-1.amzn2023.0.4.x86_64.rp 4.1 MB/s | 105 kB    00:00
(10/21): krb5-libs-1.21-3.amzn2023.0.4.x86_64.rp 16 MB/s | 758 kB    00:00
(11/21): libmount-2.37.4-1.amzn2023.0.4.x86_64.r 7.9 MB/s | 132 kB    00:00
(12/21): libnghttp2-1.59.0-3.amzn2023.0.1.x86_64 5.6 MB/s | 79 kB    00:00
(13/21): libsmartcols-2.37.4-1.amzn2023.0.4.x86_ 4.4 MB/s | 62 kB    00:00
(14/21): libcurl-minimal-8.5.0-1.amzn2023.0.4.x8 7.1 MB/s | 275 kB    00:00
(15/21): libuuid-2.37.4-1.amzn2023.0.4.x86_64.rp 1.1 MB/s | 26 kB    00:00
```

```
(16/21): python3-3.9.16-1.amzn2023.0.8.x86_64.rpm 1.5 MB/s | 27 kB      00:00
(17/21): python3-dnf-4.14.0-1.amzn2023.0.5.noarch 19 MB/s | 409 kB     00:00
(18/21): system-release-2023.5.20240708-1.amzn20 1.9 MB/s | 28 kB      00:00
(19/21): yum-4.14.0-1.amzn2023.0.5.noarch.rpm   1.6 MB/s | 32 kB      00:00
(20/21): openssl-libs-3.0.8-1.amzn2023.0.12.x86_ 26 MB/s | 2.2 MB     00:00
(21/21): python3-libs-3.9.16-1.amzn2023.0.8.x86_ 59 MB/s | 7.3 MB     00:00
```

```
-----
Total                                     34 MB/s | 14 MB      00:00
```

Running transaction check

Transaction check succeeded.

Running transaction test

Transaction test succeeded.

Running transaction

```
Preparing      : 1/1
Upgrading      : glibc-common-2.34-52.amzn2023.0.10.x86_64 1/42
Upgrading      : glibc-minimal-langpack-2.34-52.amzn2023.0.10.x86_64 2/42
Running scriptlet: glibc-2.34-52.amzn2023.0.10.x86_64 3/42
Upgrading      : glibc-2.34-52.amzn2023.0.10.x86_64 3/42
Running scriptlet: glibc-2.34-52.amzn2023.0.10.x86_64 3/42
Upgrading      : libuuid-2.37.4-1.amzn2023.0.4.x86_64 4/42
Upgrading      : openssl-libs-1:3.0.8-1.amzn2023.0.12.x86_64 5/42
Upgrading      : krb5-libs-1.21-3.amzn2023.0.4.x86_64 6/42
Upgrading      : libblkid-2.37.4-1.amzn2023.0.4.x86_64 7/42
Running scriptlet: libblkid-2.37.4-1.amzn2023.0.4.x86_64 7/42
Upgrading      : expat-2.5.0-1.amzn2023.0.4.x86_64 8/42
Upgrading      : python3-3.9.16-1.amzn2023.0.8.x86_64 9/42
Upgrading      : python3-libs-3.9.16-1.amzn2023.0.8.x86_64 10/42
Upgrading      : libnghttp2-1.59.0-3.amzn2023.0.1.x86_64 11/42
Upgrading      : libcurl-minimal-8.5.0-1.amzn2023.0.4.x86_64 12/42
Upgrading      : system-release-2023.5.20240708-1.amzn2023.noarch 13/42
Upgrading      : amazon-linux-repo-cdn-2023.5.20240708-1.amzn2023.no 14/42
Upgrading      : dnf-data-4.14.0-1.amzn2023.0.5.noarch 15/42
Upgrading      : python3-dnf-4.14.0-1.amzn2023.0.5.noarch 16/42
Upgrading      : dnf-4.14.0-1.amzn2023.0.5.noarch 17/42
Running scriptlet: dnf-4.14.0-1.amzn2023.0.5.noarch 17/42
Upgrading      : yum-4.14.0-1.amzn2023.0.5.noarch 18/42
Upgrading      : curl-minimal-8.5.0-1.amzn2023.0.4.x86_64 19/42
Upgrading      : libmount-2.37.4-1.amzn2023.0.4.x86_64 20/42
Upgrading      : libsmartcols-2.37.4-1.amzn2023.0.4.x86_64 21/42
Cleanup        : yum-4.14.0-1.amzn2023.0.4.noarch 22/42
Running scriptlet: dnf-4.14.0-1.amzn2023.0.4.noarch 23/42
Cleanup        : dnf-4.14.0-1.amzn2023.0.4.noarch 23/42
Running scriptlet: dnf-4.14.0-1.amzn2023.0.4.noarch 23/42
Cleanup        : python3-dnf-4.14.0-1.amzn2023.0.4.noarch 24/42
```

```

Cleanup      : amazon-linux-repo-cdn-2023.4.20240319-1.amzn2023.no 25/42
Cleanup      : libmount-2.37.4-1.amzn2023.0.3.x86_64           26/42
Cleanup      : curl-minimal-8.5.0-1.amzn2023.0.2.x86_64       27/42
Cleanup      : libcurl-minimal-8.5.0-1.amzn2023.0.2.x86_64    28/42
Cleanup      : krb5-libs-1.21-3.amzn2023.0.3.x86_64           29/42
Cleanup      : libblkid-2.37.4-1.amzn2023.0.3.x86_64          30/42
Cleanup      : libnghttp2-1.57.0-1.amzn2023.0.1.x86_64        31/42
Cleanup      : libsmartcols-2.37.4-1.amzn2023.0.3.x86_64      32/42
Cleanup      : system-release-2023.4.20240319-1.amzn2023.noarch 33/42
Cleanup      : dnf-data-4.14.0-1.amzn2023.0.4.noarch           34/42
Cleanup      : python3-3.9.16-1.amzn2023.0.6.x86_64           35/42
Cleanup      : python3-libs-3.9.16-1.amzn2023.0.6.x86_64      36/42
Cleanup      : openssl-libs-1:3.0.8-1.amzn2023.0.11.x86_64    37/42
Cleanup      : libuuid-2.37.4-1.amzn2023.0.3.x86_64           38/42
Cleanup      : expat-2.5.0-1.amzn2023.0.3.x86_64              39/42
Cleanup      : glibc-2.34-52.amzn2023.0.8.x86_64              40/42
Cleanup      : glibc-minimal-langpack-2.34-52.amzn2023.0.8.x86_64 41/42
Cleanup      : glibc-common-2.34-52.amzn2023.0.8.x86_64       42/42
Running scriptlet: glibc-common-2.34-52.amzn2023.0.8.x86_64 42/42
Verifying    : amazon-linux-repo-cdn-2023.5.20240708-1.amzn2023.no 1/42
Verifying    : amazon-linux-repo-cdn-2023.4.20240319-1.amzn2023.no 2/42
Verifying    : curl-minimal-8.5.0-1.amzn2023.0.4.x86_64       3/42
Verifying    : curl-minimal-8.5.0-1.amzn2023.0.2.x86_64       4/42
Verifying    : dnf-4.14.0-1.amzn2023.0.5.noarch                5/42
Verifying    : dnf-4.14.0-1.amzn2023.0.4.noarch                6/42
Verifying    : dnf-data-4.14.0-1.amzn2023.0.5.noarch           7/42
Verifying    : dnf-data-4.14.0-1.amzn2023.0.4.noarch           8/42
Verifying    : expat-2.5.0-1.amzn2023.0.4.x86_64              9/42
Verifying    : expat-2.5.0-1.amzn2023.0.3.x86_64             10/42
Verifying    : glibc-2.34-52.amzn2023.0.10.x86_64            11/42
Verifying    : glibc-2.34-52.amzn2023.0.8.x86_64             12/42
Verifying    : glibc-common-2.34-52.amzn2023.0.10.x86_64     13/42
Verifying    : glibc-common-2.34-52.amzn2023.0.8.x86_64     14/42
Verifying    : glibc-minimal-langpack-2.34-52.amzn2023.0.10.x86_64 15/42
Verifying    : glibc-minimal-langpack-2.34-52.amzn2023.0.8.x86_64 16/42
Verifying    : krb5-libs-1.21-3.amzn2023.0.4.x86_64          17/42
Verifying    : krb5-libs-1.21-3.amzn2023.0.3.x86_64          18/42
Verifying    : libblkid-2.37.4-1.amzn2023.0.4.x86_64         19/42
Verifying    : libblkid-2.37.4-1.amzn2023.0.3.x86_64         20/42
Verifying    : libcurl-minimal-8.5.0-1.amzn2023.0.4.x86_64   21/42
Verifying    : libcurl-minimal-8.5.0-1.amzn2023.0.2.x86_64   22/42
Verifying    : libmount-2.37.4-1.amzn2023.0.4.x86_64         23/42
Verifying    : libmount-2.37.4-1.amzn2023.0.3.x86_64         24/42
Verifying    : libnghttp2-1.59.0-3.amzn2023.0.1.x86_64       25/42

```

```

Verifying      : libnghttp2-1.57.0-1.amzn2023.0.1.x86_64      26/42
Verifying      : libsmartcols-2.37.4-1.amzn2023.0.4.x86_64   27/42
Verifying      : libsmartcols-2.37.4-1.amzn2023.0.3.x86_64   28/42
Verifying      : libuuid-2.37.4-1.amzn2023.0.4.x86_64       29/42
Verifying      : libuuid-2.37.4-1.amzn2023.0.3.x86_64       30/42
Verifying      : openssl-libs-1:3.0.8-1.amzn2023.0.12.x86_64 31/42
Verifying      : openssl-libs-1:3.0.8-1.amzn2023.0.11.x86_64 32/42
Verifying      : python3-3.9.16-1.amzn2023.0.8.x86_64       33/42
Verifying      : python3-3.9.16-1.amzn2023.0.6.x86_64       34/42
Verifying      : python3-dnf-4.14.0-1.amzn2023.0.5.noarch    35/42
Verifying      : python3-dnf-4.14.0-1.amzn2023.0.4.noarch    36/42
Verifying      : python3-libs-3.9.16-1.amzn2023.0.8.x86_64  37/42
Verifying      : python3-libs-3.9.16-1.amzn2023.0.6.x86_64  38/42
Verifying      : system-release-2023.5.20240708-1.amzn2023.noarch 39/42
Verifying      : system-release-2023.4.20240319-1.amzn2023.noarch 40/42
Verifying      : yum-4.14.0-1.amzn2023.0.5.noarch           41/42
Verifying      : yum-4.14.0-1.amzn2023.0.4.noarch           42/42

```

Upgraded:

```

amazon-linux-repo-cdn-2023.5.20240708-1.amzn2023.noarch
curl-minimal-8.5.0-1.amzn2023.0.4.x86_64
dnf-4.14.0-1.amzn2023.0.5.noarch
dnf-data-4.14.0-1.amzn2023.0.5.noarch
expat-2.5.0-1.amzn2023.0.4.x86_64
glibc-2.34-52.amzn2023.0.10.x86_64
glibc-common-2.34-52.amzn2023.0.10.x86_64
glibc-minimal-langpack-2.34-52.amzn2023.0.10.x86_64
krb5-libs-1.21-3.amzn2023.0.4.x86_64
libblkid-2.37.4-1.amzn2023.0.4.x86_64
libcurl-minimal-8.5.0-1.amzn2023.0.4.x86_64
libmount-2.37.4-1.amzn2023.0.4.x86_64
libnghttp2-1.59.0-3.amzn2023.0.1.x86_64
libsmartcols-2.37.4-1.amzn2023.0.4.x86_64
libuuid-2.37.4-1.amzn2023.0.4.x86_64
openssl-libs-1:3.0.8-1.amzn2023.0.12.x86_64
python3-3.9.16-1.amzn2023.0.8.x86_64
python3-dnf-4.14.0-1.amzn2023.0.5.noarch
python3-libs-3.9.16-1.amzn2023.0.8.x86_64
system-release-2023.5.20240708-1.amzn2023.noarch
yum-4.14.0-1.amzn2023.0.5.noarch

```

Complete!

bash-5.2#

Per scoprire AL2023 gli aggiornamenti, esegui una o più delle seguenti operazioni:

- Esegui il comando `dnf check-update`. Questo verifica la presenza di eventuali aggiornamenti non applicati nella versione di Amazon Linux a cui sei bloccato. Questo potrebbe mostrare degli aggiornamenti se hai aggiornato solo il `system-release` pacchetto, spostando la versione del repository su cui è bloccata l'istanza ma non applicando nessuno degli aggiornamenti disponibili in essa.
- Sottoscrivi l'argomento SNS per l'aggiornamento del repository Amazon Linux (`arn:aws:sns:us-east-1:137112412989:amazon-linux-2023-ami-updates`). Per ulteriori informazioni, consulta [Sottoscrizione a un argomento di Amazon SNS](#) nella Guida per lo Sviluppatore di Amazon Simple Notification Service.
- Consulta regolarmente le [note di AL2023 rilascio](#).
- Scopri le nuove versioni di [Verifica della disponibilità di versioni più recenti del repository con `dnf check-release-update`](#).

Important

Nuove versioni di AL2023 contenenti aggiornamenti di sicurezza vengono rilasciate frequentemente. Assicurati di tenerti aggiornato sulle patch di sicurezza pertinenti.

Riavvio automatico del servizio dopo gli aggiornamenti (di sicurezza)

Amazon Linux ora viene fornito con il pacchetto [smart-restart](#). `smart-restart` riavvia i servizi `systemd` sugli aggiornamenti di sistema ogni volta che un pacchetto viene installato o eliminato utilizzando il gestore di pacchetti del sistema. Ciò si verifica ogni volta che `dnf (update | upgrade | downgrade)` viene eseguito.

`smart-restart` utilizza il `needs-restarting` pacchetto from `dnf-utils` e un meccanismo di `denylisting` personalizzato per determinare quali servizi devono essere riavviati e se è consigliato il riavvio del sistema. Se si consiglia il riavvio del sistema, viene generato un file marker di suggerimento per il riavvio (`/run/smart-restart/reboot-hint-marker`

Per installare **smart-restart**

Eseguite il DNF comando seguente (come fareste con qualsiasi altro pacchetto).

```
$ sudo dnf install smart-restart
```

Dopo l'installazione, le transazioni successive attiveranno la `smart-restart` logica.

Lista di rifiuto

`Smart-restart` può essere richiesto di bloccare il riavvio di determinati servizi. I servizi bloccati non contribuiranno alla decisione se è necessario un riavvio. Per bloccare servizi aggiuntivi, aggiungete un file con il suffisso `-denylist` in `/etc/smart-restart-conf.d/` come illustrato nell'esempio seguente.

```
$ cat /etc/smart-restart-conf.d/custom-denylist
# Some comments
myservice.service
```

Note

Tutti `*-denylist` i file vengono letti e valutati quando si decide se è necessario un riavvio.

Ganci personalizzati

Oltre al denylisting, `smart-restart` fornisce un meccanismo per eseguire script personalizzati prima e dopo i tentativi di riavvio del servizio. Gli script personalizzati possono essere utilizzati per eseguire manualmente le fasi di preparazione o per informare gli altri componenti del riavvio residuo o completato.

Tutti gli script vengono inseriti `/etc/smart-restart-conf.d/` con il suffisso `-pre-restart` o `-post-restart` vengono eseguiti. Se l'ordine è importante, aggiungete un numero a tutti gli script per garantire l'ordine di esecuzione, come illustrato nell'esempio seguente.

```
$ ls /etc/smart-restart-conf.d/*-pre-restart
001-my-script-pre-restart
002-some-other-script-pre-restart
```

Quando è necessario il riavvio per applicare gli aggiornamenti di sicurezza?

In alcune situazioni, Amazon Linux richiede un riavvio per applicare gli aggiornamenti:

- Gli aggiornamenti al pacchetto del kernel Linux richiedono un riavvio per attivare il nuovo kernel con gli ultimi aggiornamenti di sicurezza. Il livpatching del kernel può consentire di posticipare gli aggiornamenti di sicurezza per un periodo di tempo limitato. Per ulteriori informazioni, consultare [Kernel Live Patching attivo AL2023](#)
- Sulle istanze EC2 Metal, Amazon Linux fornisce aggiornamenti del microcodice (tramite il `microcode_ctl` pacchetto per Intel CPUs e il `amd-ucode-firmware` pacchetto per le CPU AMD). Questi aggiornamenti del microcodice verranno attivati solo ai riavvii successivi delle istanze. Per le istanze EC2 virtualizzate, il [sistema AWS Nitro](#) sottostante gestisce gli aggiornamenti del microcodice per te.
- Alcuni servizi `systemd` in esecuzione funzioneranno correttamente solo dopo un riavvio completo del sistema. Il `smart-restart` meccanismo ti informerà su tali situazioni lasciando dei suggerimenti per il riavvio. Per informazioni, consulta [Riavvio automatico del servizio dopo gli aggiornamenti \(di sicurezza\)](#).

Avvio di un'istanza con la versione più recente del repository abilitata

Puoi aggiungere comandi DNF a uno script di dati utente per controllare quali pacchetti RPM vengono installati su un'AMI Amazon Linux al momento dell'avvio. Nell'esempio seguente, viene utilizzato uno script di dati utente per assicurarsi che su ogni istanza avviata con lo script di dati utente siano installati gli stessi aggiornamenti del pacchetto.

```
#!/bin/bash
dnf upgrade --releasever=2023.0.20230210
# Additional setup and install commands below
dnf install httpd php7.4 mysql80
```

È necessario eseguire questo script come utente con privilegi avanzati (`root`). Per farlo, esegui il comando seguente.

```
$ sudo sh -c "bash nameofscript.sh"
```

Per ulteriori informazioni, consulta [Dati utente e script di shell nella Guida](#) per l'utente di Amazon EC2.

Note

Invece di utilizzare uno script di dati utente, avvia l'AMI Amazon Linux più recente o un'AMI personalizzata basata sull'AMI Amazon Linux. L'AMI Amazon Linux più recente dispone di tutti gli aggiornamenti necessari installati ed è configurata in modo che punti a una particolare versione del repository.

Ottenere informazioni di supporto per i pacchetti

AL2023 incorpora molti progetti software open source diversi. Ciascuno di questi progetti è gestito indipendentemente da Amazon Linux e prevede release e end-of-support pianificazioni diverse. Per fornirti informazioni specifiche su Amazon Linux in relazione a questi diversi pacchetti, il plugin `supportinfo` DNF fornisce i metadati relativi a un pacchetto. Nell'esempio seguente, il comando `dnf supportinfo` restituisce i metadati per il pacchetto `glibc`.

```
$ sudo dnf supportinfo --pkg glibc
Last metadata expiration check: 0:07:56 ago on Wed Mar 1 23:21:49 2023.
Name           : glibc
Version        : 2.34-52.amzn2023.0.2
State          : installed
Support Status  : supported
Support Periods : from 2023-03-15      : supported
                : from 2028-03-15      : unsupported
Support Statement : Amazon Linux 2023 End Of Life
Link           : https://aws.amazon.com/amazon-linux-ami/faqs/
Other Info      : This is the support statement for AL2023. The
                ...: end of life of Amazon Linux 2023 would be March 2028.
                ...: From this point, the Amazon Linux 2023 packages (listed
                ...: below) will no longer, receive any updates from AWS.
```

Le informazioni sul supporto dei Package sono disponibili anche nella sezione delle [istruzioni di supporto](#) delle [Note di AL2023 rilascio](#).

Verifica della disponibilità di versioni più recenti del repository con `dnf check-release-update`

In un AL2023 caso, è possibile utilizzare l'IDNFutilità per gestire gli archivi e applicare pacchetti aggiornati. RPM Questi pacchetti sono disponibili nei repository Amazon Linux. Puoi usare il

comando DNF `dnf check-release-update` per verificare la presenza di nuove versioni del repository DNF.

Note

AL2023 per impostazione predefinita, le immagini dei contenitori non includono il `dnf check-release-update` comando.

```
$ dnf check-release-update
```

```
No such command: check-release-update. Please use /usr/bin/dnf --help
It could be a DNF plugin command, try: "dnf install 'dnf-command(check-release-update)'"
```

Quando `dnf install 'dnf-command(check-release-update)'` viene eseguito, `dnf` installerà il pacchetto che fornisce il `check-release-update` comando, che è il `dnf-plugin-release-notification` pacchetto. Nell'esempio seguente, viene data l'argomento `dnf` affinché abbia un output silenzioso.

```
$ dnf -y -q install 'dnf-command(check-release-update)'
```

```
Installed:
  dnf-plugin-release-notification-1.2-1.amzn2023.0.2.noarch
```

In ambienti non containerizzati come un'istanza Amazon EC2, il `check-release-update` comando è incluso per impostazione predefinita.

```
$ sudo dnf check-release-update
```

```
WARNING:
```

```
A newer release of "Amazon Linux" is available.
```

```
Available Versions:
```

```
Version 2023.0.20230210:
```

```
Run the following command to update to 2023.0.20230210:
```

```
dnf upgrade --releasever=2023.0.20230210
```

```
Release notes:
```

```
https://docs.aws.amazon.com/linux/al2023/release-notes/relnotes.html
```

Questa operazione restituisce un elenco completo di tutte le versioni più recenti dei repository DNF disponibili. Se non viene restituito nulla, significa che DNF è attualmente configurato per utilizzare la versione più recente disponibile. La versione del pacchetto `system-release` attualmente installata imposta la variabile `releasever` DNF. Per controllare la versione attuale del repository, esegui il comando seguente.

```
$ rpm -q system-release --qf "%{VERSION}\n"
```

Quando si eseguono transazioni di pacchetti DNF (ad esempio comandi di installazione, aggiornamento o rimozione), un messaggio di avviso segnala eventuali nuove versioni del repository. Ad esempio, se si installa il `httpd` pacchetto su un'istanza lanciata da una versione precedente di AL2023, viene restituito il seguente output.

```
$ sudo dnf install httpd -y
Last metadata expiration check: 0:16:52 ago on Wed Mar 1 23:21:49 2023.
Dependencies resolved.
=====
Package                Arch   Version                Repository   Size
=====
Installing:
httpd                   x86_64 2.4.54-3.amzn2023.0.4  amazonlinux 46 k
Installing dependencies:
apr                     x86_64 1.7.2-2.amzn2023.0.2  amazonlinux 129 k
apr-util                x86_64 1.6.3-1.amzn2023.0.1  amazonlinux 98 k
generic-logos-httpd
noarch                 18.0.0-12.amzn2023.0.3 amazonlinux 19 k
httpd-core              x86_64 2.4.54-3.amzn2023.0.4  amazonlinux 1.3 M
httpd-filesystem       noarch 2.4.54-3.amzn2023.0.4  amazonlinux 13 k
httpd-tools             x86_64 2.4.54-3.amzn2023.0.4  amazonlinux 80 k
libbrotli               x86_64 1.0.9-4.amzn2023.0.2  amazonlinux 315 k
mailcap                 noarch 2.1.49-3.amzn2023.0.3  amazonlinux 33 k
Installing weak dependencies:
apr-util-openssl       x86_64 1.6.3-1.amzn2023.0.1  amazonlinux 17 k
mod_http2               x86_64 1.15.24-1.amzn2023.0.3 amazonlinux 152 k
mod_lua                 x86_64 2.4.54-3.amzn2023.0.4  amazonlinux 60 k

Transaction Summary
=====
Install 12 Packages

Total download size: 2.3 M
Installed size: 6.8 M
```

Downloading Packages:

```
(1/12): apr-util-openssl-1.6.3-1.am 212 kB/s | 17 kB    00:00
(2/12): apr-1.7.2-2.amzn2023.0.2.x8 1.1 MB/s | 129 kB    00:00
(3/12): httpd-core-2.4.54-3.amzn202 8.9 MB/s | 1.3 MB    00:00
(4/12): mod_http2-1.15.24-1.amzn202 1.9 MB/s | 152 kB    00:00
(5/12): apr-util-1.6.3-1.amzn2023.0 1.7 MB/s | 98 kB     00:00
(6/12): mod_lua-2.4.54-3.amzn2023.0 1.4 MB/s | 60 kB     00:00
(7/12): httpd-2.4.54-3.amzn2023.0.4 1.5 MB/s | 46 kB     00:00
(8/12): libbrotli-1.0.9-4.amzn2023. 4.4 MB/s | 315 kB    00:00
(9/12): mailcap-2.1.49-3.amzn2023.0 753 kB/s | 33 kB     00:00
(10/12): httpd-tools-2.4.54-3.amzn2 978 kB/s | 80 kB     00:00
(11/12): httpd-filesystem-2.4.54-3. 210 kB/s | 13 kB     00:00
(12/12): generic-logos-httpd-18.0.0 439 kB/s | 19 kB     00:00
```

```
-----
Total                               6.6 MB/s | 2.3 MB    00:00
```

Running transaction check

Transaction check succeeded.

Running transaction test

Transaction test succeeded.

Running transaction

```
Preparing      :                               1/1
Installing     : apr-1.7.2-2.amzn2023.0.2.x86_64 1/12
Installing     : apr-util-openssl-1.6.3-1.amzn2023.0.1. 2/12
Installing     : apr-util-1.6.3-1.amzn2023.0.1.x86_64 3/12
Installing     : mailcap-2.1.49-3.amzn2023.0.3.noarch 4/12
Installing     : httpd-tools-2.4.54-3.amzn2023.0.4.x86_ 5/12
Installing     : generic-logos-httpd-18.0.0-12.amzn2023 6/12
Running scriptlet: httpd-filesystem-2.4.54-3.amzn2023.0.4 7/12
Installing     : httpd-filesystem-2.4.54-3.amzn2023.0.4 7/12
Installing     : httpd-core-2.4.54-3.amzn2023.0.4.x86_6 8/12
Installing     : mod_http2-1.15.24-1.amzn2023.0.3.x86_6 9/12
Installing     : libbrotli-1.0.9-4.amzn2023.0.2.x86_64 10/12
Installing     : mod_lua-2.4.54-3.amzn2023.0.4.x86_64 11/12
Installing     : httpd-2.4.54-3.amzn2023.0.4.x86_64 12/12
Running scriptlet: httpd-2.4.54-3.amzn2023.0.4.x86_64 12/12
Verifying      : apr-1.7.2-2.amzn2023.0.2.x86_64 1/12
Verifying      : apr-util-openssl-1.6.3-1.amzn2023.0.1. 2/12
Verifying      : httpd-core-2.4.54-3.amzn2023.0.4.x86_6 3/12
Verifying      : mod_http2-1.15.24-1.amzn2023.0.3.x86_6 4/12
Verifying      : apr-util-1.6.3-1.amzn2023.0.1.x86_64 5/12
Verifying      : mod_lua-2.4.54-3.amzn2023.0.4.x86_64 6/12
Verifying      : libbrotli-1.0.9-4.amzn2023.0.2.x86_64 7/12
Verifying      : httpd-2.4.54-3.amzn2023.0.4.x86_64 8/12
Verifying      : httpd-tools-2.4.54-3.amzn2023.0.4.x86_ 9/12
```

```
Verifying      : mailcap-2.1.49-3.amzn2023.0.3.noarch    10/12
Verifying      : httpd-filesystem-2.4.54-3.amzn2023.0.4  11/12
Verifying      : generic-logos-httpd-18.0.0-12.amzn2023  12/12
```

Installed:

```
apr-1.7.2-2.amzn2023.0.2.x86_64
apr-util-1.6.3-1.amzn2023.0.1.x86_64
apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64
generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch
httpd-2.4.54-3.amzn2023.0.4.x86_64
httpd-core-2.4.54-3.amzn2023.0.4.x86_64
httpd-filesystem-2.4.54-3.amzn2023.0.4.noarch
httpd-tools-2.4.54-3.amzn2023.0.4.x86_64
libbrotli-1.0.9-4.amzn2023.0.2.x86_64
mailcap-2.1.49-3.amzn2023.0.3.noarch
mod_http2-1.15.24-1.amzn2023.0.3.x86_64
mod_lua-2.4.54-3.amzn2023.0.4.x86_64
```

Complete!

Aggiunta, abilitazione o disabilitazione di nuovi repository

Warning

Aggiungi solo repository progettati per essere utilizzati con AL2023. Sebbene i repository progettati per altre distribuzioni possano funzionare oggi, non c'è alcuna garanzia che continueranno a farlo con qualsiasi aggiornamento del pacchetto AL2023 o con il repository non progettato per essere utilizzato. AL2023

Per installare un pacchetto da un repository diverso da quello predefinito di Amazon Linux, dovrai configurare il sistema di gestione dei DNF pacchetti per sapere dove si trova il repository.

Per dnf informazioni su un repository di pacchetti, aggiungi le informazioni del repository a un file di configurazione per quel repository nella directory. /etc/yum.repos.d/ Molti repository di terze parti forniscono il contenuto del file di configurazione o un pacchetto installabile che include il file di configurazione.

Note

Sebbene i repository possano essere configurati direttamente nel `/etc/dnf/dnf.conf` file, questa operazione non è consigliata. Si consiglia di configurare ogni repository nel proprio file in `/etc/yum.repos.d/`

Per scoprire quali repository sono attualmente abilitati, puoi eseguire il seguente comando:

```
$ dnf repolist all --verbose
```

```
Loaded plugins: builddep, changelog, config-manager, copr, debug, debuginfo-install,
download, generate_completion_cache, groups-manager, needs-restarting, playground,
release-notification, repoclosure, repodiff, repograph, repomanage, reposync,
supportinfo
```

```
DNF version: 4.12.0
```

```
cachedir: /var/cache/dnf
```

```
Last metadata expiration check: 0:00:02 ago on Wed Mar 1 23:40:15 2023.
```

```
Repo-id           : amazonlinux
```

```
Repo-name         : Amazon Linux 2023 repository
```

```
Repo-status       : enabled
```

```
Repo-revision     : 1677203368
```

```
Repo-updated      : Fri Feb 24 01:49:28 2023
```

```
Repo-pkgs         : 12632
```

```
Repo-available-pkgs: 12632
```

```
Repo-size         : 12 G
```

```
Repo-mirrors      : https://al2023-repos-us-west-2-de612dc2.s3.dualstack.us-
west-2.amazonaws.com/core/mirrors/2023.0.20230222/x86_64/mirror.list
```

```
Repo-baseurl     : https://al2023-repos-us-west-2-de612dc2.s3.dualstack.us-
west-2.amazonaws.com/core/guids/
```

```
cf9296325a6c46ff40c775a8e2d632c4c3fd9d9164014ce3304715d61b90ca8e/x86_64/
```

```
: (0 more)
```

```
Repo-expire       : 172800 second(s) (last: Wed Mar 1 23:40:15
```

```
: 2023)
```

```
Repo-filename     : /etc/yum.repos.d/amazonlinux.repo
```

```
Repo-id           : amazonlinux-debuginfo
```

```
Repo-name         : Amazon Linux 2023 repository - Debug
```

```
Repo-status       : disabled
```

```
Repo-mirrors      : https://al2023-repos-us-west-2-de612dc2.s3.dualstack.us-
west-2.amazonaws.com/core/mirrors/2023.0.20230222/debuginfo/x86_64/mirror.list
```

```
Repo-expire       : 21600 second(s) (last: unknown)
```

```
Repo-filename     : /etc/yum.repos.d/amazonlinux.repo
```

```

Repo-id           : amazonlinux-source
Repo-name        : Amazon Linux 2023 repository - Source packages
Repo-status      : disabled
Repo-mirrors     : https://al2023-repos-us-west-2-de612dc2.s3.dualstack.us-
west-2.amazonaws.com/core/mirrors/2023.0.20230222/SRPMS/mirror.list
Repo-expire      : 21600 second(s) (last: unknown)
Repo-filename    : /etc/yum.repos.d/amazonlinux.repo

Repo-id           : kernel-livepatch
Repo-name        : Amazon Linux 2023 Kernel Livepatch repository
Repo-status      : disabled
Repo-mirrors     : https://al2023-repos-us-west-2-de612dc2.s3.dualstack.us-
west-2.amazonaws.com/kernel-livepatch/mirrors/al2023/x86_64/mirror.list
Repo-expire      : 172800 second(s) (last: unknown)
Repo-filename    : /etc/yum.repos.d/kernel-livepatch.repo

Repo-id           : kernel-livepatch-source
Repo-name        : Amazon Linux 2023 Kernel Livepatch repository -
                  : Source packages
Repo-status      : disabled
Repo-mirrors     : https://al2023-repos-us-west-2-de612dc2.s3.dualstack.us-
west-2.amazonaws.com/kernel-livepatch/mirrors/al2023/SRPMS/mirror.list
Repo-expire      : 21600 second(s) (last: unknown)
Repo-filename    : /etc/yum.repos.d/kernel-livepatch.repo
Total packages: 12632

```

Note

Se non aggiungi il flag di opzione `--verbose`, l'output include solo le informazioni `Repo-id`, `Repo-name` e `Repo-status`.

Per aggiungere un repository `yum` alla directory `/etc/yum.repos.d`:

1. Cerca la posizione del file `.repo`. In questo esempio, il file `.repo` è disponibile in <https://www.example.com/repository.repo>.
2. Aggiungi il repository con il comando `dnf config-manager`.

```

$ sudo dnf config-manager --add-repo https://www.example.com/repository.repo
Loaded plugins: priorities, update-motd, upgrade-helper

```

```
adding repo from: https://www.example.com/repository.repo
grabbing file https://www.example.com/repository.repo to /etc/
yum.repos.d/repository.repo
repository.repo | 4.0 kB 00:00
repo saved to /etc/yum.repos.d/repository.repo
```

Dopo aver installato un repository, devi abilitarlo come descritto nella procedura seguente.

Per abilitare un yum repository in `/etc/yum.repos.d`, usa il `dnf config-manager` comando con `--enable` bandiera e *repository* nome.

```
$ sudo dnf config-manager --enable repository
```

Note

Per disabilitare un repository, usa la stessa sintassi del comando, ma sostituendo `--enable` con `--disable` nel comando.

Aggiunta di repository con cloud-init

Oltre ad aggiungere un repository utilizzando il precedente metodo, puoi anche aggiungere un nuovo repository utilizzando il framework `cloud-init`.

Per aggiungere un nuovo repository di pacchetti, si consiglia l'uso del seguente modello. Valuta la possibilità di salvare il file in locale.

```
#cloud-config
yum_repos:
  repository.repo:
    baseurl: https://www.example.com/
    enabled: true
    gpgcheck: true
    gpgkey: file:///etc/pki/rpm-gpg/RPM-GPG-KEY-EXAMPLE
    name: Example Repository
```

Note

Un vantaggio dell'utilizzo di `cloud-init` consiste nella possibilità di aggiungere una sezione `packages:` al file di configurazione. In questa sezione, puoi includere i nomi dei pacchetti

che desideri installare. Puoi installare i pacchetti dal repository predefinito o dal nuovo repository aggiunto nel file `cloud-config`.

Per informazioni più specifiche sulla struttura del file YAML, consulta la pagina dedicata all'[aggiunta di un repository YUM](#) nella documentazione di `cloud-init`.

Dopo aver configurato il file in formato YAML, puoi eseguirlo nel framework `cloud-init` nell'interfaccia AWS CLI. Assicurati di includere l'opzione `--userdata` e il nome del file `.yaml` per chiamare le operazioni desiderate.

```
$ aws ec2 run-instances \  
  --image-id \  
    resolve:ssm:/aws/service/ami-amazon-linux-latest/aL2023-ami-kernel-default-x86_64 \  
  --instance-type m5.xlarge \  
  --region us-east-1 \  
  --key-name aws-key-us-east-1 \  
  --security-group-ids sg-004a7650 \  
  --user-data file://cloud-config.yaml
```

Kernel Live Patching attivo AL2023

Puoi usare Kernel Live Patching per applicare vulnerabilità di sicurezza specifiche e patch di bug critici AL2023 a un kernel Linux in esecuzione senza riavviare o interrompere le applicazioni in esecuzione. Inoltre, Kernel Live Patching può contribuire a migliorare la disponibilità dell'applicazione applicando queste correzioni fino al riavvio del sistema.

AWS rilascia due tipi di patch live del kernel per: AL2023

- **Aggiornamenti di sicurezza:** includono aggiornamenti per CVE (Common Vulnerabilities and Exposures) di Linux. Questi aggiornamenti sono in genere classificati come importanti o critici utilizzando le classificazioni di Amazon Linux Security Advisory. In alcuni casi, AWS potrebbe fornire aggiornamenti prima dell'assegnazione di un CVE. In questi casi, le patch potrebbero apparire come correzioni di bug. Le patch live del kernel per gli aggiornamenti di sicurezza vengono fornite con la massima diligenza possibile: non tutte le patch importanti o critiche CVEs possono essere disponibili come patch live per una determinata versione del kernel. Per determinare se un CVE specifico è stato risolto da una patch live, [consulta gli avvisi di sicurezza di Amazon Linux](#).
- **Correzioni di bug:** include correzioni di bug critici e problemi di stabilità non associati a CVEs

AWS fornisce patch live del kernel per una versione del AL2023 kernel per un massimo di 3 mesi dopo il suo rilascio. Dopo tale periodo, è necessario eseguire l'aggiornamento a una versione del kernel successiva per continuare a ricevere patch live del kernel.

AL2023 le patch live del kernel sono rese disponibili come pacchetti RPM firmati nei repository esistenti. AL2023 Le patch possono essere installate su singole istanze utilizzando i flussi di lavoro del gestore di pacchetti DNF esistenti. In alternativa, possono essere installati su un gruppo di istanze gestite utilizzando AWS Systems Manager.

Kernel Live Patching on AL2023 viene fornito senza costi aggiuntivi.

Argomenti

- [Limitazioni](#)
- [Configurazioni e prerequisiti supportati](#)
- [Utilizzo di Kernel Live Patching](#)

Limitazioni

Durante l'applicazione di una patch live del kernel, non è possibile eseguire l'ibernazione, utilizzare strumenti di debug avanzati (come SystemTap, kprobes e strumenti basati su eBPF) né accedere ai file di output `fttrace` utilizzati dall'infrastruttura Kernel Live Patching.

Note

A causa di limitazioni tecniche, alcuni problemi non possono essere risolti con live patching. Per questo motivo, queste correzioni non verranno incluse nel pacchetto kernel live patch ma solo nell'aggiornamento del pacchetto kernel nativo. È possibile installare il pacchetto kernel nativo e [aggiornare e riavviare](#) il sistema per attivare le patch come al solito.

Configurazioni e prerequisiti supportati

Kernel Live Patching è supportato sulle istanze Amazon EC2 e sulle macchine virtuali locali in esecuzione. AL2023

Per utilizzare Kernel Live Patching su AL2023, devi usare quanto segue:

- Un'architettura `x86_64` o `ARM64` a 64 bit

- Versione del kernel o 6.16.12

Requisiti delle politiche

Per scaricare pacchetti dai repository AL2023, Amazon EC2 deve accedere ai bucket Amazon S3 di proprietà del servizio. Se utilizzi un endpoint Amazon Virtual Private Cloud (VPC) per Amazon S3 nel tuo ambiente, assicurati che la policy degli endpoint VPC consenta l'accesso a tali bucket pubblici. La tabella seguente descrive il bucket Amazon S3 a cui Amazon EC2 potrebbe dover accedere per Kernel Live Patching.

ARN di bucket S3	Description
arn:aws:s3:::al2023-repos- <i>de612dc2</i> /* <i>region</i>	Bucket Amazon S3 contenente repository AL2023

Utilizzo di Kernel Live Patching

È possibile abilitare e utilizzare Kernel Live Patching su singole istanze utilizzando la riga di comando sull'istanza stessa. In alternativa, è possibile abilitare e utilizzare Kernel Live Patching su un gruppo di istanze gestite utilizzando AWS Systems Manager.

Le sezioni seguenti spiegano come abilitare e utilizzare Kernel Live Patching su singole istanze utilizzando la riga di comando.

Per ulteriori informazioni sull'abilitazione e l'utilizzo di Kernel Live Patching su un gruppo di istanze gestite, consulta [Use Kernel Live Patching sulle istanze nella Guida](#) per l'utente. AL2023 AWS Systems Manager

Argomenti

- [Abilitazione di Kernel Live Patching](#)
- [Visualizzazione delle patch live del kernel disponibili](#)
- [Applicazione delle patch live del kernel](#)
- [Visualizzazione delle patch live del kernel applicate](#)
- [Disabilitazione di Kernel Live Patching](#)

Abilitazione di Kernel Live Patching

Kernel Live Patching è disabilitato per impostazione predefinita su AL2023. Per utilizzare l'applicazione di patch live, è necessario installare il plugin DNF per Kernel Live Patching e abilitare la funzionalità di applicazione di patch live.

Per abilitare Kernel Live Patching

1. Le patch live del kernel sono disponibili per AL2023 tutte le versioni del kernel. 6.1 Per controllare la versione del kernel, eseguire il seguente comando.

```
$ sudo dnf list kernel
```

2. Installare il plugin DNF per Kernel Live Patching.

```
$ sudo dnf install -y kpatch-dnf
```

3. Abilitare il plugin DNF per Kernel Live Patching.

```
$ sudo dnf kernel-livepatch -y auto
```

Questo comando installa anche l'ultima versione dell'RPM della patch live del kernel dai repository configurati.

4. Per confermare che il plugin DNF per Kernel Live Patching è stato installato correttamente, eseguire il seguente comando.

Quando si abilita Kernel Live Patching, viene automaticamente applicata una RPM della patch live del kernel vuota. Se Kernel Live Patching è stato abilitato correttamente, questo comando restituisce un elenco che include l'RPM iniziale vuoto del kernel live patch (e un altro RPM che configura il repository DNF contenente le livpatch).

```
$ sudo rpm -qa | grep kernel-livepatch
kernel-livepatch-repo-s3-2023.7.20250428-0.amzn2023.noarch
kernel-livepatch-6.1.134-150.224-1.0-0.amzn2023.x86_64
```

5. Installare il pacchetto kpatch.

```
$ sudo dnf install -y kpatch-runtime
```

6. Aggiornare il servizio kpatch se è stato precedentemente installato.

```
$ sudo dnf upgrade kpatch-runtime
```

7. Avviare il servizio kpatch. Questo servizio carica tutte le patch live del kernel dopo l'inizializzazione o l'avvio.

```
$ sudo systemctl enable kpatch.service && sudo systemctl start kpatch.service
```

Visualizzazione delle patch live del kernel disponibili

Gli avvisi di sicurezza di Amazon Linux vengono pubblicati nel Centro di Sicurezza Amazon Linux. Per ulteriori informazioni sugli avvisi AL2023 di sicurezza, inclusi gli avvisi per le patch live del kernel, consulta [Amazon](#) Linux Security Center. Le patch live del kernel sono precedute da ALASLIVEPATCH. Centro di Sicurezza Amazon Linux potrebbe non elencare le patch live del kernel che risolvono i bug.

Puoi anche scoprire le patch live del kernel disponibili per ricevere avvisi e utilizzare la riga di comando. CVEs

Per elencare tutte le patch live del kernel disponibili per le consulenze

Utilizza il seguente comando.

```
$ sudo dnf updateinfo list
```

```
Last metadata expiration check: 1:06:23 ago on Mon 13 Feb 2023 09:28:19 PM UTC.  
ALAS2LIVEPATCH-2021-123    important/Sec. kernel-  
livepatch-6.1.12-17.42-1.0-4.amzn2023.x86_64  
ALAS2LIVEPATCH-2022-124    important/Sec. kernel-  
livepatch-6.1.12-17.42-1.0-3.amzn2023.x86_64
```

Per elencare tutte le patch live del kernel disponibili per CVEs

Utilizza il seguente comando.

```
$ sudo dnf updateinfo list cves
```

```
Last metadata expiration check: 1:07:26 ago on Mon 13 Feb 2023 09:28:19 PM UTC.  
CVE-2022-0123    important/Sec. kernel-livepatch-6.1.12-17.42-1.0-4.amzn2023.x86_64  
CVE-2022-3210    important/Sec. kernel-livepatch-6.1.12-17.42-1.0-3.amzn2023.x86_64
```

Applicazione delle patch live del kernel

Le patch live del kernel vengono applicate utilizzando il gestore di pacchetti DNF nello stesso modo in cui si applicano aggiornamenti regolari. Il plugin DNF per Kernel Live Patching gestisce le patch live del kernel disponibili per l'applicazione.

Tip

Si consiglia di aggiornare regolarmente il kernel utilizzando Kernel Live Patching per garantire che riceva correzioni di sicurezza specifiche fino al riavvio del sistema. Controlla anche se sono state rese disponibili correzioni aggiuntive al pacchetto kernel nativo che non possono essere distribuite come patch live e [aggiorna e riavvia l'aggiornamento](#) del kernel in questi casi.

Puoi scegliere di applicare una patch live del kernel specifica o applicare qualsiasi patch live del kernel disponibile insieme ai normali aggiornamenti di sicurezza.

Per applicare una patch live del kernel specifica

1. Ottenere la versione della patch live del kernel utilizzando uno dei comandi descritti in [Visualizzazione delle patch live del kernel disponibili](#).
2. Applica la kernel live patch per il tuo kernel. AL2023

```
$ sudo dnf install kernel-livepatch-kernel_version-package_version.amzn2023.x86_64
```

Ad esempio, il comando seguente applica una patch live del kernel per la versione del kernel AL2023 6.1.12-17.42

```
$ sudo dnf install kernel-livepatch-6.1.12-17.42-1.0-4.amzn2023.x86_64
```

Per applicare eventuali patch live del kernel disponibili insieme ai normali aggiornamenti di sicurezza

Utilizzare il seguente comando.

```
$ sudo dnf upgrade --security
```

Omettere l'opzione `--security` per includere correzioni di bug.

Important

- La versione del kernel non viene aggiornata dopo l'applicazione delle patch live del kernel. La versione viene aggiornata alla nuova versione solo dopo il riavvio dell'istanza.
- Un AL2023 kernel riceve le patch live del kernel per 3 mesi. Dopo questo periodo, non vengono rilasciate nuove patch live del kernel per tale versione del kernel.
- Per continuare a ricevere patch live del kernel dopo 3 mesi, è necessario riavviare l'istanza per passare alla nuova versione del kernel. L'istanza continua a ricevere le patch live del kernel per i successivi 3 mesi dopo l'aggiornamento.
- Per controllare la finestra di supporto per la versione del kernel, esegui il seguente comando:

```
$ sudo dnf kernel-livepatch support
```

```
The current version of the Linux kernel you are running will no longer receive  
live patches after 2025-07-22.
```

Visualizzazione delle patch live del kernel applicate

Per visualizzare le patch live del kernel applicate

Utilizzare il seguente comando.

```
$ sudo kpatch list
```

```
Loaded patch modules:
```

```
livepatch_CVE_2022_36946 [enabled]
```

```
Installed patch modules:
```

```
livepatch_CVE_2022_36946 (6.1.57-29.131.amzn2023.x86_64)
```

```
livepatch_CVE_2022_36946 (6.1.57-30.131.amzn2023.x86_64)
```

Il comando restituisce un elenco delle patch live del kernel dell'aggiornamento di sicurezza caricato e installato. Di seguito è riportato un output di esempio.

Note

Una singola patch live del kernel può includere e installare più patch live.

Disabilitazione di Kernel Live Patching

Se non è più necessario utilizzare Kernel Live Patching, puoi disabilitarla in qualsiasi momento.

- Disabilita l'uso di livepatches:

1. Disabilita il plugin:

```
$ sudo dnf kernel-livepatch manual
```

2. Disabilita il servizio kpatch:

```
$ sudo systemctl disable --now kpatch.service
```

- Rimuovi completamente gli strumenti livepatch:

1. Rimuovi il plugin:

```
$ sudo dnf remove kpatch-dnf
```

2. Rimuovi kpatch-runtime:

```
$ sudo dnf remove kpatch-runtime
```

3. Rimuovi tutte le livepatches installate:

```
$ sudo dnf remove kernel-livepatch\*
```

Aggiornamento del kernel Linux su AL2023

Argomenti

- [Versioni del kernel Linux su AL2023](#)
- [AL2023 Aggiornamento a una versione più recente del kernel](#)
- [AL2023 kernels - Domande frequenti](#)

Versioni del kernel Linux su AL2023

AL2023 include regolarmente nuove versioni del kernel basate sulle versioni Long-Term Support (LTS) del kernel Linux.

AL2023 è stato originariamente rilasciato nel marzo 2023 con kernel 6.1.

Nell'aprile 2025, è AL2023 stato aggiunto il supporto per il kernel Linux 6.12. Questo kernel ha aggiunto nuove funzionalità tra cui la pianificazione EEVDF, il I/O supporto passthrough FUSE, una nuova API Futex e miglioramenti in eBPF. Kernel 6.12 consente inoltre a un programma userspace di proteggersi in fase di esecuzione utilizzando stack shadow dello spazio utente e sigillando la memoria.

Nel marzo 2026, è stato aggiunto il supporto per il kernel Linux 6.18 AL2023 . Il kernel 6.18 aggiornato apporta ulteriori miglioramenti nel supporto del processore, nella virtualizzazione, nella sicurezza e nelle prestazioni. Le caratteristiche più importanti includono funzionalità IOMMU migliorate su tutte le architetture e Attack Vector Controls per la gestione delle mitigazioni delle vulnerabilità della CPU. I miglioramenti delle prestazioni derivano dalle ottimizzazioni della crittografia con operazioni FSCRYPT più veloci, dai miglioramenti della gestione della memoria e dall'introduzione di Sheaves come nuovo livello di caching basato su array opt-in per CPU.

AL2023 Aggiornamento a una versione più recente del kernel

A partire da giugno 2026, AL2023 aggiornerà il kernel predefinito ogni anno. Il [al2023-ami-kernel-default](#) set AMIs verrà aggiornato all'ultima versione del kernel LTS, in modo che le istanze appena lanciate presentino automaticamente la nuova versione del kernel: questo è il modo più semplice per rimanere aggiornati sulle ultime correzioni di sicurezza e sui miglioramenti delle prestazioni.

Se preferisci scegliere una versione del kernel specifica, puoi eseguire AL2023 con kernel 6.12 o 6.18 selezionando un'AMI con il kernel desiderato preinstallato o aggiornando un'istanza AL2023 EC2 esistente.

Esecuzione di un' AL2023 AMI con una versione del kernel specifica

Puoi scegliere di eseguire un' AL2023 AMI con un kernel specifico preinstallato tramite la console AWS o interrogando SSM per parametri specifici. Le chiavi SSM con cui eseguire la query iniziano seguite da una delle `/aws/service/ami-amazon-linux-latest/`

Per il kernel 6.12

- `al2023-ami-kernel-6.12-arm64` per l'architettura arm64
- `al2023-ami-minimal-kernel-6.12-arm64` per l'architettura arm64 (AMI minima)
- `al2023-ami-kernel-6.12-x86_64` per l'architettura x86_64

- `al2023-ami-minimal-kernel-6.12-x86_64` per l'architettura `x86_64` (AMI minima)

Per il kernel 6.18

- `al2023-ami-kernel-6.18-arm64` per l'architettura `arm64`
- `al2023-ami-minimal-kernel-6.18-arm64` per l'architettura `arm64` (AMI minima)
- `al2023-ami-kernel-6.18-x86_64` per l'architettura `x86_64`
- `al2023-ami-minimal-kernel-6.18-x86_64` per l'architettura `x86_64` (AMI minima)

[Avvio tramite il parametro SSM e AL2023 AWS CLI](#) Per i dettagli sulla selezione, vedere. AL2023 AMIs

Aggiornamento di un' AL2023 istanza a un kernel più recente

È possibile aggiornare sul posto un' AL2023 istanza in esecuzione al kernel 6.12 o 6.18 con i seguenti passaggi:

1. Rileva il kernel corrente e imposta la versione di destinazione:

```
# Automatically detect current kernel version BEFORE upgrade
$ CURRENT_KERNEL=$(uname -r)
$ SOURCE_VERSION=""

$ if [[ $CURRENT_KERNEL == *"6.12"* ]]; then
    SOURCE_VERSION="6.12"
else
    SOURCE_VERSION=""
fi

# Save the source version to a persistent location for use after reboot
$ echo "${SOURCE_VERSION}" | sudo tee /var/lib/source_kernel_version > /dev/null
```

```
# Set your target version (change this to your desired kernel: 6.12 or 6.18)
$ TARGET_VERSION="6.12"
```

```
$ echo "Current kernel: ${SOURCE_VERSION:-6.1}"
$ echo "Upgrading to kernel ${TARGET_VERSION}"
```

2. Installa il pacchetto kernel di destinazione:

```
$ sudo dnf install -y kernel${TARGET_VERSION}
```

3. Scarica l'ultima versione del pacchetto kernel di destinazione:

```
$ version=$(rpm -q --qf '%{version}-%{release}.%{arch}\n' kernel${TARGET_VERSION} |
sort -V | tail -1)
```

4. Rendi il nuovo kernel il tuo kernel predefinito:

```
$ sudo grubby --set-default "/boot/vmlinuz-$version"
```

5. Riavvia il sistema:

```
$ sudo reboot
```

6. Disinstalla il kernel precedente:

```
# Read the source kernel version from the saved file
$ SOURCE_VERSION=$(sudo cat /var/lib/source_kernel_version)

# Uninstall the source kernel
$ sudo dnf remove -y kernel${SOURCE_VERSION}
```

7. Sostituisci i pacchetti kernel aggiuntivi con i loro equivalenti del kernel di destinazione:

```
# Set your target version (change this to your desired kernel: 6.12 or 6.18)
$ TARGET_VERSION="6.12"
```

```
$ declare -A pkgs
$ pkgs=(
[bpftool${SOURCE_VERSION}]=bpftool${TARGET_VERSION}
[kernel${SOURCE_VERSION}-debuginfo]=kernel${TARGET_VERSION}-debuginfo
[kernel${SOURCE_VERSION}-debuginfo-common]=kernel${TARGET_VERSION}-debuginfo-common
[kernel${SOURCE_VERSION}-headers]=kernel${TARGET_VERSION}-headers
[kernel${SOURCE_VERSION}-libbpf]=kernel${TARGET_VERSION}-libbpf
[kernel${SOURCE_VERSION}-libbpf-devel]=kernel${TARGET_VERSION}-libbpf-devel
[kernel${SOURCE_VERSION}-libbpf-static]=kernel${TARGET_VERSION}-libbpf-static
[kernel${SOURCE_VERSION}-modules-extra-common]=kernel${TARGET_VERSION}-modules-
extra-common
[kernel${SOURCE_VERSION}-tools]=kernel${TARGET_VERSION}-tools
[kernel${SOURCE_VERSION}-tools-devel]=kernel${TARGET_VERSION}-tools-devel
```

```
[perf${SOURCE_VERSION}]=perf${TARGET_VERSION}
[python3-perf${SOURCE_VERSION}]=python3-perf${TARGET_VERSION}
)
$ for pkg in "${!pkgs[@]}"; do
  rpm -q $pkg && sudo dnf -y swap $pkg "${pkgs["$pkg"]}" ;
done
```

8. (Opzionale) Disinstalla kernel-devel per la versione precedente del kernel:

```
$ rpm -q kernel${SOURCE_VERSION}-devel && sudo dnf remove -y kernel
${SOURCE_VERSION}-devel
```

Eseguire il downgrade a una versione precedente del kernel

Se in qualsiasi momento hai bisogno di tornare a una versione precedente del kernel, usa i seguenti passaggi:

1. Rileva il kernel corrente e imposta la versione di destinazione:

```
# Automatically detect current kernel version BEFORE downgrade
$ CURRENT_KERNEL=$(uname -r)
$ SOURCE_VERSION=""

$ if [[ $CURRENT_KERNEL == *"6.12"* ]]; then
  SOURCE_VERSION="6.12"
elif [[ $CURRENT_KERNEL == *"6.18"* ]]; then
  SOURCE_VERSION="6.18"
fi

# Save the source version to a persistent location for use after reboot
$ echo "${SOURCE_VERSION}" | sudo tee /var/lib/source_kernel_version > /dev/null
```

```
# Set your target version (change this to your desired kernel)
# Use "" for kernel 6.1, "6.12" for kernel 6.12
$ TARGET_VERSION=""
```

```
$ echo "Downgrading from kernel ${SOURCE_VERSION:-6.1} to kernel
${TARGET_VERSION:-6.1}"
```

2. Sostituisci i pacchetti kernel aggiuntivi con i loro equivalenti del kernel di destinazione:

```
$ declare -A pkgs
$ pkgs=(
  [bpftool${TARGET_VERSION}]=bpftool${SOURCE_VERSION}
  [kernel${TARGET_VERSION}-debuginfo]=kernel${SOURCE_VERSION}-debuginfo
  [kernel${TARGET_VERSION}-debuginfo-common]=kernel${SOURCE_VERSION}-debuginfo-common
  [kernel${TARGET_VERSION}-headers]=kernel${SOURCE_VERSION}-headers
  [kernel${TARGET_VERSION}-libbpf]=kernel${SOURCE_VERSION}-libbpf
  [kernel${TARGET_VERSION}-libbpf-devel]=kernel${SOURCE_VERSION}-libbpf-devel
  [kernel${TARGET_VERSION}-libbpf-static]=kernel${SOURCE_VERSION}-libbpf-static
  [kernel${TARGET_VERSION}-modules-extra-common]=kernel${SOURCE_VERSION}-modules-extra-common
  [kernel${TARGET_VERSION}-tools]=kernel${SOURCE_VERSION}-tools
  [kernel${TARGET_VERSION}-tools-devel]=kernel${SOURCE_VERSION}-tools-devel
  [perf${TARGET_VERSION}]=perf${SOURCE_VERSION}
  [python3-perf${TARGET_VERSION}]=python3-perf${SOURCE_VERSION}
)
$ for pkg in "${!pkgs[@]}"; do
  rpm -q "${pkgs["$pkg"]}" && sudo dnf -y swap "${pkgs["$pkg"]}" $pkg ;
done
```

3. Installa il pacchetto kernel di destinazione:

```
$ sudo dnf install -y kernel${TARGET_VERSION}
```

4. Scarica l'ultima versione del pacchetto kernel di destinazione:

```
$ version=$(rpm -q --qf '%{version}-%{release}.%{arch}\n' kernel${TARGET_VERSION} |
  sort -V | tail -1)
```

5. Rendi il kernel di destinazione il tuo kernel predefinito:

```
$ sudo grubby --set-default "/boot/vmlinuz-$version"
```

6. Riavvia il sistema:

```
$ sudo reboot
```

7. Disinstalla il kernel sorgente:

```
# Read the source kernel version from the saved file
$ SOURCE_VERSION=$(sudo cat /var/lib/source_kernel_version)

# Uninstall the source kernel
$ sudo dnf remove -y kernel${SOURCE_VERSION}
```

AL2023 kernels - Domande frequenti

1. Devo riavviare il sistema dopo un aggiornamento del kernel?

Ogni modifica al kernel in esecuzione richiede un riavvio.

2. Come faccio a mantenere i kernel up-to-date su più istanze?

Amazon Linux non fornisce strutture per gestire flotte di istanze. Ti consigliamo di applicare patch a flotte di grandi dimensioni utilizzando strumenti come [AWS Systems Manager](#).

3. Come posso verificare quale versione del kernel sto utilizzando in questo momento?

Esegui questo comando sulla tua AL2023 istanza:

```
$ uname -r
```

4. Quale kernel mi AL2023 consiglia di usare?

Si consiglia di eseguire l'aggiornamento all'ultimo AL2023 kernel 6.18 mentre tutti gli altri AL2023 kernel sono ancora supportati. Si consiglia ai clienti di testare i propri carichi di lavoro prima dell'aggiornamento.

5. Le mie applicazioni esistenti funzioneranno con qualsiasi AL2023 kernel?

AL2023 supporta un kernel più recente (6.12 o 6.18) allo stesso modo del kernel 6.1. Le applicazioni funzioneranno e i miglioramenti avverranno sotto il cofano. I clienti devono in ogni caso testare i propri carichi di lavoro specifici prima di passare a un kernel più recente.

6. Come posso installare gli header del kernel, i pacchetti di sviluppo e i moduli aggiuntivi per il kernel 6.12 o 6.18?

Per favore esegui:

```
$ version=$(uname -r | grep -oP '^\\d+\\.\\d+')  
$ sudo dnf install -y kernel${version}-modules-extra-$(uname -r) kernel${version}-  
headers-$(uname -r) kernel${version}-devel-$(uname -r)
```

7. Per quanto tempo saranno supportati i kernel 6.12 e 6.18?

I kernel 6.12 e 6.18 saranno supportati fino alla fine del ciclo di vita prevista di Amazon Linux 2023, ovvero il 30 giugno 2029.

Guida introduttiva alla programmazione dei runtime su AL2023

AL2023 fornisce diverse versioni di alcuni runtime linguistici. Lavoriamo con progetti upstream che supportano più versioni contemporaneamente. Scopri come installare e gestire questi pacchetti dotati di versioni con nomi utilizzando il comando `dnf` per cercare e installare questi pacchetti.

I seguenti argomenti descrivono come esiste ogni ecosistema linguistico. AL2023

Argomenti

- [CC++, e Fortran nel AL2 2023](#)
- [.NET nel AL2023](#)
- [Gonel AL2023](#)
- [Javanel AL2 2023](#)
- [Node.js nel AL2 2023](#)
- [Perl nel AL2 2023](#)
- [PHP nel AL2 2023](#)
- [Python nel AL2023](#)
- [Rubynel AL2023](#)
- [Rust nel AL2023](#)
- [Swift nel AL2023](#)
- [TypeScript nel AL2 2023](#)

CC++, e Fortran nel AL2 2023

AL2023 include sia la GNU Compiler Collection (GCC) che il Clang frontend per LLVM (Low Level Virtual Machine).

La versione principale di 023 GCC rimarrà costante per tutta la durata di vita di 023. AL2 Le versioni minori apportano correzioni di bug e potrebbero essere incluse nelle versioni AL2 023. Altre correzioni di bug, prestazioni e sicurezza potrebbero essere riportate nella versione principale disponibile nella 023. GCC AL2

AL2023 include la versione 11 di GCC come compilatore predefinito con i frontend C (gcc), C++ (g++) e Fortran (gfortran). Inoltre, AL2023 fornisce la GCC versione 14 come compilatore alternativo opzionale che può essere installato insieme alla versione predefinita.

AL2023 non abilita i frontend Ada (gnat), Go (gcc-go), Objective-C o Objective-C++.

I flag predefiniti del compilatore con cui è stato creato AL2023 includono i flag di ottimizzazione e rafforzamento. Per creare il tuo codice con GCC, ti consigliamo di includere flag di ottimizzazione e rafforzamento.

Note

Quando `gcc --version` viene richiamato, viene visualizzata una stringa di versione come `gcc (GCC) 11.3.1 20221121 (Red Hat 11.3.1-4)`. Red Hat si riferisce al [ramo del fornitore GCC](#) su cui si basa il pacchetto di Amazon Linux GCC. In base all'URL di segnalazione dei bug mostrato da `gcc --help`, tutte le segnalazioni di bug e le richieste di supporto devono essere indirizzate ad Amazon Linux.

Per maggiori informazioni su alcune delle modifiche a lungo termine in questo ramo del fornitore, come la `__GNUC_RH_RELEASE__` macro, vedi i sorgenti dei pacchetti [Fedora](#).

Per ulteriori informazioni sulla toolchain di base, vedere [Pacchetti di toolchain principali glibc, gcc, binutils](#)

Per ulteriori informazioni su AL2023 e sulla sua relazione con altre distribuzioni Linux, vedere [Relazione con Fedora](#)

Per ulteriori informazioni sulla variazione della tripletta del compilatore in AL2023 rispetto a, vedere [AL2023 Tripletta del compilatore](#)

Argomenti

- [GCC14](#)
- [Confronto tra le versioni linguistiche standard](#)

GCC14

AL2023 fornisce GCC 14 come compilatore opzionale che può essere installato insieme all'11 predefinito. GCC14 include le funzionalità e le ottimizzazioni più recenti del linguaggio, il che lo rende adatto a progetti che richiedono il supporto degli standard C, C++ o Fortran più recenti.

Per installare GCC 14, usa il seguente comando:

```
sudo dnf install gcc14 gcc14-c++ gcc14-gfortran
```

I GCC 14 compilatori vengono installati con nomi di comando specifici della versione per evitare conflitti con l'11 predefinito: GCC

- `gcc14-gcc`- compilatore C
- `gcc14-g++`- compilatore C++
- `gcc14-gfortran`- compilatore Fortran

Esempio di utilizzo:

```
gcc14-gcc -o myprogram myprogram.c
gcc14-g++ -o mycppprogram mycppprogram.cpp
gcc14-gfortran -o myfortranprogram myfortranprogram.f90
```

È possibile verificare la versione installata eseguendo:

```
gcc14-gcc --version
```

Verranno visualizzate informazioni sulla versione simili a: `gcc14-gcc (GCC) 14.2.1 20250110 (Red Hat 14.2.1-7)`

Note

La GCC versione 11 e la GCC 14 possono essere installate contemporaneamente sullo stesso sistema. I `gfortran` comandi predefiniti `gcc` e continueranno a utilizzare GCC 11, mentre GCC 14 è accessibile tramite i comandi specifici della versione. `g++`

Confronto tra le versioni linguistiche standard

La tabella seguente confronta le versioni standard del linguaggio predefinito tra diverse versioni di Amazon Linux e versioni del GCC compilatore:

Versione di Amazon Linux	C Standard (impostazione predefinita)	Standard C++ (impostazione predefinita)	Norma Fortran
AL2 con GCC 7 (impostazione predefinita)	C11 (201112L)	C+14 (2014-02 L)	Fortran 2008
AL2 con GCC 10 (opzionale)	C17/C18 (201710L)	C++ 14 (2014-02 L)	Fortran 2008
AL2023 con GCC 11 (impostazione predefinita)	C17/C18 (201710L)	C++ 17 (201703 L)	Fortran 2008
AL2023 con GCC 14 (opzionale)	C17/C18 (201710L)	C++ 17 (201703 L)	Fortran 2008

Principali miglioramenti per versione: GCC

- GCC10 vs GCC 7: standard C predefinito aggiornato da C11 a C17/C18, aggiunto supporto per le funzionalità di C++20 e funzionalità di ottimizzazione migliorate.
- GCC11 vs GCC 10: standard C++ predefinito aggiornato da C++14 a C++17, supporto C++20 migliorato e funzionalità sperimentali di C++23.
- GCC14 vs GCC 11: aggiunto il supporto completo dello standard C23, funzionalità C++23 migliorate, ottimizzazione migliorata e migliore conformità agli standard.

Standard linguistici supportati:

- Standard C: tutte le versioni supportano C90, C99, C11 e C17/C18. GCC10+ supporta C2x (bozza C23), mentre 14 fornisce il supporto completo per C23. GCC
- Standard C++: tutte le versioni supportano C++98, C++03, C++11, C++14, C++17 e C++20. GCC11+ fornisce supporto sperimentale per C++23, con 14 che offrono funzionalità C++23 avanzate. GCC
- Standard Fortran: tutte le versioni supportano principalmente Fortran 2008, con diversi livelli di funzionalità di Fortran 2018 a seconda della versione. GCC

Note

Sebbene gli standard predefiniti rimangano coerenti tra GCC 11 e 14, GCC 14 offre un supporto delle funzionalità linguistiche notevolmente migliorato, una migliore ottimizzazione, una diagnostica migliorata e un'implementazione più completa degli standard più recenti quando richiesto esplicitamente utilizzando i flag. `-std=`

.NET nel AL2023

AL2023 attualmente fornisce il [.NET](#) runtime e l'SDK, versioni 8.0, 9.0 e 10.0. .NET6.0 è stato raggiunto end-of-life a novembre 2024 e non è più supportato. Ogni versione è supportata per lo stesso periodo di tempo della versione upstream. .NET Per ulteriori informazioni, vedere [Package support statements](#).

Migrazione a versioni più recenti .NET

La .NET comunità upstream fornisce documentazione sulla migrazione per lo spostamento di:

- [a .NET 10](#)
- [fino a .NET 9](#)
- [fino a .NET 8](#)

Installazione .NET su AL2023

È possibile installare contemporaneamente più versioni del .NET runtime su AL2023. Tuttavia, i `dotnet-host` pacchetti `dotnet` and sono condivisi tra le versioni SDK, quindi solo una versione SDK può possedere il punto di ingresso `/usr/bin/dotnet` CLI alla volta. L'installazione di una versione SDK diversa aggiornerà il `dotnet` comando condiviso a quella versione.

Per installare una versione specifica dell'.NET SDK, usa il `dnf` comando:

```
sudo dnf install dotnet-sdk-10.0
```

Sostituisci `10.0` con la versione desiderata (8.0, 9.0 o 10.0).

Per installare solo il runtime senza l'SDK:

```
sudo dnf install dotnet-runtime-9.0
```

Gonel AL2023

Potresti voler creare il tuo codice scritto [Gosu](#) su Amazon Linux e potresti voler utilizzare una toolchain fornita con AL2023. Analogamente AL2, AL2023 aggiornerà la Go toolchain per tutta la durata del sistema operativo. Questo potrebbe avvenire in risposta a qualsiasi CVE nella toolchain che forniamo o come parte di un rilascio trimestrale.

Go è una lingua che si muove relativamente velocemente. Potrebbe esserci una situazione in cui le applicazioni esistenti scritte Go devono adattarsi alle nuove versioni della Go toolchain. Per ulteriori informazioni su Go, vedere [Go1 and the Future of Go Programs](#).

Sebbene AL2023 incorporerà nuove versioni della Go toolchain nel corso del suo ciclo di vita, ciò non sarà in linea con le versioni precedenti. Go Pertanto, l'utilizzo della Go toolchain fornita AL2023 potrebbe non essere adatto se si desidera creare Go codice utilizzando funzionalità all'avanguardia del linguaggio e della libreria standard. Go

Durante la durata di AL2023, le versioni precedenti dei pacchetti non vengono rimosse dai repository. Se è necessaria una Go toolchain precedente, potete scegliere di rinunciare alle correzioni di bug e di sicurezza delle Go toolchain più recenti e installare una versione precedente dai repository utilizzando gli stessi meccanismi disponibili per qualsiasi RPM.

Se desideri creare il tuo Go codice AL2023, puoi utilizzare la Go toolchain inclusa nella AL2023 confezione sapendo che questa toolchain potrebbe andare avanti per tutta la vita di AL2023

AL2023 Funzioni Lambda scritte in Go

Durante la Go compilazione in codice nativo, Lambda Go tratta come un runtime personalizzato. Puoi usare il `provided.al2023` runtime per distribuire Go funzioni su AL2023 Lambda.

Per ulteriori informazioni, consulta [Creazione di funzioni Lambda Go nella Guida](#) per gli AWS Lambda sviluppatori.

Javanel AL2 2023

AL2023 fornisce diverse versioni di [Amazon Corretto](#) per supportare carichi di lavoro Java basati. Tutti i pacchetti Java basati inclusi in AL2 023 sono creati con Amazon Corretto 17

Corretto è una build dell'Open Java Development Kit (OpenJDK) con supporto a lungo termine da Amazon. Corretto è certificato utilizzando il Java Technical Compatibility Kit (TCK) per garantire che soddisfi lo standard Java SE e sia disponibile su LinuxWindows, macOS.

È disponibile un pacchetto [Amazon Corretto](#) per ciascuno dei pacchetti Corretto 1.8.0, Corretto 11 e Corretto 17.

Ogni versione di Corretto nella versione AL2 023 è supportata per lo stesso periodo di tempo della versione Corretto o fino alla fine del ciclo di vita della AL2 023, a seconda di quale dei due eventi si verifichi per prima. Per ulteriori informazioni, consulta le [dichiarazioni di supporto dei pacchetti Amazon Linux](#) e [Amazon Corretto FAQs](#).

Node.js nel AL2 2023

[Node.js](#) in AL2 023 è rappresentato dalle versioni 20, 22 e 24. Amazon Linux segue la [pianificazione del supporto](#) upstream e lo stato del supporto di qualsiasi versione di Node.js può sempre essere verificato nella [pagina di stato del supporto Package](#). Tutte le Node.js versioni supportate hanno uno spazio dei nomi e possono essere installate contemporaneamente sullo stesso sistema. Il namespacing garantisce che ogni Node.js installazione sia unica all'interno del file system. Ciò si ottiene rinominando le directory e i file chiave in base alla versione runtime. I nomi eseguibili effettivi saranno simili a `node-{MAJOR_VERSION}` o `npm - {MAJOR_VERSION}`. Tuttavia, può essere attiva solo una Node.js versione alla volta. Questa versione attiva fornisce le directory e i nomi di file predefiniti, come `node`, `npm` o `/usr/lib/node_modules`, indirizzandoli al runtime attualmente attivo.

Ciò si ottiene utilizzando le funzionalità dello strumento delle alternative. È importante ricordare che i nomi eseguibili predefiniti sono virtuali e possono cambiare in qualsiasi momento quando si fa riferimento a una Node.js versione installata diversa. Questa flessibilità consente al software che utilizza il nodo nello shebang di selezionare la versione desiderata quando viene richiamata. Tuttavia, quando Node.js è richiesta una versione specifica di, è possibile ottenere la persistenza della versione chiamando l'eseguibile con namespace (ad esempio, `node-20` o `node-22`), che utilizzerà sempre la versione specificata del runtime. Inoltre, gli eseguibili con namespace dello strumento `npm`, come `npm-20` o `npm-22`, sono sempre associati alla versione Node.js corrispondente, indipendentemente dal runtime attualmente attivo.

Node.js è distribuito come diversi pacchetti con namespace che iniziano con "».

`nodejs{MAJOR_VERSION}` Questi pacchetti forniscono `node`, una versione compatibile dello strumento `npm`, documentazione, librerie e altro. Ad esempio, `node` e `npm` of the Node.js 22 sono forniti rispettivamente dai `nodejs22-npm` pacchetti `nodejs22` and.

Lo strumento delle alternative fornisce un unico comando per passare da una Node.js versione all'altra. Per impostazione predefinita, le alternative sono configurate per essere in modalità auto, che utilizza le priorità per determinare la Node.js versione attualmente attiva. Tuttavia, è possibile attivare qualsiasi versione installata in qualsiasi momento. Attualmente, tutte le versioni supportate di Node.js hanno la stessa priorità, il che significa che la prima versione installata verrà attivata automaticamente.

Alcuni esempi utili di utilizzo di alternative

1. Controlla per quali alternative è configurato

```
alternatives --list
```

2. Controlla la configurazione attuale del nodo

```
alternatives --display node
```

3. Cambia la versione in modo interattivo Node.js

```
alternatives --config node
```

4. Passa alla modalità manuale e seleziona una versione specifica

```
alternatives --set node /usr/bin/node-{MAJOR_VERSION}
```

5. Torna alla modalità di selezione automatica della versione

```
alternatives --auto node
```

Perl nel AL2 2023

AL2023 fornisce la versione 5.32 di [Perl](#) linguaggio di programmazione.

Sebbene Perl ha fornito un elevato grado di compatibilità linguistica nell'ambito di Perl 5 versioni negli ultimi decenni, non si prevede che Amazon Linux si sposti da Perl 5.32 durante la versione AL2 023. Amazon Linux continuerà a utilizzare le patch di sicurezza Perl per la durata di AL2 023 in conformità con le nostre [dichiarazioni di supporto ai pacchetti](#).

Perl moduli in 023 AL2

Vari Perl i moduli sono impacchettati come RPMs in AL2 023. Sebbene ce ne siano molti Perl moduli disponibili come RPMs, Amazon Linux non mira a impacchettare tutti i possibili pacchetti Perl modulo. Moduli confezionati come RPMs potrebbero essere utilizzati da altri pacchetti RPM del sistema operativo, quindi Amazon Linux darà la priorità a tali patch di sicurezza rispetto ai puri aggiornamenti delle funzionalità.

AL2023 include anche CPAN Perl gli sviluppatori possono usare il gestore di pacchetti idiomatico per Perl moduli.

PHPnel AL2 2023

AL2023 attualmente fornisce il linguaggio di [PHP](#) programmazione, le versioni 8.1, 8.2, 8.3, 8.4 e 8.5. Ogni versione è supportata per lo stesso periodo di tempo della versione upstream. PHP Per ulteriori informazioni, vedere [Package support statements](#).

Migrazione da versioni PHP precedenti

La PHP comunità upstream ha messo insieme una documentazione completa sulla migrazione per lo spostamento di:

- [da PHP 8.4.x a 8.5.x PHP](#)
- [da 8.3.x a 8.4.x PHPPHP](#)
- [da 8.2.x a 8.3.x PHPPHP](#)
- [da 8.1.x a 8.2.x PHPPHP](#)
- [da 8.0.x a 8.1.x PHPPHP](#)

AL2 include PHP 8.0, 8.1 e 8.2 per `amazon-linux-extras` consentire un facile percorso di aggiornamento a 023. AL2

Migrazione da PHP versioni 7.x

Note

[Il PHPprogetto mantiene un elenco e una pianificazione delle versioni supportate, nonché un elenco di rami non supportati.](#)

Quando AL2 023 è stato rilasciato, tutte le versioni 7.x e 5.x di 023 non [PHP](#)erano supportate dalla PHP community e non erano incluse come opzioni in 023. AL2

La PHP comunità upstream ha messo insieme una [documentazione completa sulla migrazione per passare alla versione 8.0 dalla 7.4](#). PHP PHP In combinazione con la documentazione a cui si fa riferimento nella sezione precedente sulla migrazione alla versione PHP 8.1 e PHP 8.2, è possibile migrare l'applicazione basata su una versione moderna. PHP PHP

Note

AL2 include PHP 7,1, 7,2, 7,3 e 7,4 pollici. `amazon-linux-extras` È importante notare che tutti questi Extra sono end-of-life e non sono garantiti ulteriori aggiornamenti di sicurezza.

PHPmoduli nel 2023 AL2

AL2023 include molti PHP moduli inclusi in PHP Core. AL2023 non mira a includere tutti i pacchetti nella [PHPExtension Community Library \(PECL\)](#).

Pythonnel AL2023

AL2023 rimosso Python 2.7 e tutti i componenti che lo richiedono Python sono ora scritti per funzionare con Python 3.

AL2023 rende disponibile Python 3 `/usr/bin/python3` per mantenere la compatibilità con il codice del cliente, così come il codice Python fornito con AL2023, che rimarrà Python 3.9 per tutta la vita di. AL2023

La versione di python a cui `/usr/bin/python3` punta è considerata il sistema Python e AL2023 per questo Python è la 3.9.

AL2023 attualmente fornisce versioni più recenti diPython, incluse le versioni 3.11, 3.12, 3.13 e 3.14. Ogni versione è supportata per lo stesso periodo di tempo della versione upstream. Python Per ulteriori informazioni, vedere [Package support statements](#).

PythonÈ possibile installare contemporaneamente più versioni di AL2023. Sebbene `/usr/bin/python3` sarà sempre Python 3.9, ogni versione di Python ha uno spazio dei nomi e può essere trovata in base al numero di versione. Ad esempio, se è installato `python3.11`, `/usr/bin/`

python3.11 esisterà insieme a `/usr/bin/python3.9` e al symlink `/usr/bin/python3` a `/usr/bin/python3.9`.

Note

Non modificate il punto a cui punta il `/usr/bin/python3` collegamento simbolico perché ciò potrebbe compromettere la funzionalità di base di AL2023

Migrazione a versioni più recenti Python

La Python comunità upstream fornisce la documentazione sulla migrazione per lo spostamento di:

- [a Python 3.14](#)
- [a Python 3,13](#)
- [a Python 3,12](#)
- [a Python 3.11](#)

Pythonmoduli in AL2023

Vari Python moduli sono impacchettati come RPMs in AL2023. In genere, Python i moduli verranno creati solo RPMs per la versione di sistema di Python

Rubynel AL2023

[Ruby](#) in AL2023 è rappresentato dalle versioni 3.2 e 3.4. Amazon Linux segue la pianificazione del supporto originale e lo stato del supporto di qualsiasi Ruby versione può sempre essere verificato sul sito Web di [Ruby](#). Tutte le Ruby versioni supportate hanno uno spazio dei nomi e possono essere installate contemporaneamente sullo stesso sistema. Il namespacing garantisce che ogni Ruby installazione sia unica all'interno del file system. Ciò si ottiene rinominando le directory e i file chiave in base alla versione runtime. I nomi effettivi degli eseguibili saranno simili a `ruby {MAJOR.MINOR}` (ad esempio `o.ruby3.2` `ruby3.4` `Ruby3.4` fornisce anche il binario con namespace MRI (Mats' Ruby Interpreter) `ruby3.4-mri`, che si riferisce all'implementazione di riferimento standard basata su C di Ruby. Tuttavia, può essere attiva solo una Ruby versione alla volta. Questa versione attiva fornisce le directory e i nomi di file predefiniti, come `ruby`, `gem` o `bundle`, indirizzandoli al runtime attualmente attivo.

Ciò si ottiene utilizzando le funzionalità dello strumento delle alternative. È importante ricordare che i nomi eseguibili predefiniti sono virtuali e possono cambiare in qualsiasi momento quando si fa riferimento a una Ruby versione installata diversa. Questa flessibilità consente al software che utilizza ruby in the shebang di selezionare la versione desiderata quando viene richiamata. Tuttavia, quando Ruby è richiesta una versione specifica di, è possibile ottenere la persistenza della versione chiamando l'eseguibile con namespace (ad esempio `ruby3.2` o `ruby3.4`), che utilizzerà sempre la versione specificata del runtime. Inoltre, gli eseguibili con namespace degli strumenti `gem` e `bundler`, come,,, `orruby3.2-gem`, `ruby3.4-gem` sono sempre associati alla `ruby3.2-bundler` versione corrispondente `ruby3.4-bundler`, indipendentemente dal runtime attualmente attivo. Ruby

Ruby è distribuito come diversi pacchetti con namespace che iniziano con "». `ruby{MAJOR.MINOR}` Questi pacchetti forniscono ruby, versioni compatibili degli strumenti `gem` e `bundler`, documentazione, librerie e altro. Ad esempio, il runtime core Ruby 3.2 è fornito dal `ruby3.2` pacchetto, che inserisce `ruby3.2-rubygems` (fornendo `gem`) e `ruby3.2-rubygem-bundler` (fornendo `bundle` e `bundler`) come dipendenze.

Dopo aver installato una Ruby versione, le voci relative agli strumenti complementari potrebbero essere visualizzate come nulle nella configurazione delle alternative. Questo può essere verificato `alternatives --display ruby` eseguendo. Se le voci appaiono come nulle, devono essere registrate manualmente utilizzando `alternatives --install`. Ad esempio, per registrare tutti gli strumenti complementari per Ruby 3.4:

```
sudo alternatives --install /usr/bin/gem gem /usr/bin/ruby3.4-gem 34
sudo alternatives --install /usr/bin/bundle bundle /usr/bin/ruby3.4-bundle 34
sudo alternatives --install /usr/bin/bundler bundler /usr/bin/ruby3.4-bundler 34
sudo alternatives --install /usr/bin/erb erb /usr/bin/ruby3.4-erb 34
sudo alternatives --install /usr/bin/racc racc /usr/bin/ruby3.4-racc 34
sudo alternatives --install /usr/bin/rdoc rdoc /usr/bin/ruby3.4-rdoc 34
sudo alternatives --install /usr/bin/ri ri /usr/bin/ruby3.4-ri 34
```

Il valore di priorità (ad esempio, 34 per Ruby 3.4, 32 per Ruby 3.2) deve corrispondere alla priorità utilizzata nella voce alternativa principale di ruby. Una volta registrati, gli strumenti complementari verranno gestiti automaticamente insieme all'alternativa ruby.

Lo strumento delle alternative fornisce un unico comando per passare da una Ruby versione all'altra. Per impostazione predefinita, le alternative sono configurate per essere in modalità auto, che utilizza le priorità per determinare la Ruby versione attualmente attiva. Tuttavia, è possibile attivare qualsiasi versione installata in qualsiasi momento. Attualmente, tutte le versioni supportate di Ruby hanno la stessa priorità, il che significa che la prima versione installata verrà attivata automaticamente.

Alcuni esempi utili di utilizzo di alternative

1. Controlla per quali alternative è configurato

```
alternatives --list
```

2. Controlla la configurazione attuale di ruby

```
alternatives --display ruby
```

3. Cambia la versione in modo interattivo Ruby

```
alternatives --config ruby
```

4. Passa alla modalità manuale e seleziona una versione specifica

```
alternatives --set ruby /usr/bin/ruby{MAJOR.MINOR}
```

5. Torna alla modalità di selezione automatica della versione

```
alternatives --auto ruby
```

Rustnel AL2023

Potresti voler creare codice scritto [Rust](#) su Amazon Linux e utilizzare una toolchain fornita con AL2023.

Analogamente AL2, AL2023 aggiornerà la Rust toolchain per tutta la durata del sistema operativo. Questo potrebbe avvenire in risposta a qualsiasi CVE nella toolchain che forniamo o come parte di un rilascio trimestrale.

[Rust](#) è un linguaggio che cambia in modo relativamente veloce, con nuovi rilasci all'incirca ogni sei settimane. I rilasci potrebbero aggiungere nuove funzionalità di linguaggio o di libreria standard. Sebbene AL2023 incorporerà nuove versioni della Rust toolchain nel corso del suo ciclo di vita, ciò non sarà in linea con le versioni precedenti. Rust Pertanto, l'utilizzo della Rust toolchain fornita AL2023 potrebbe non essere adatto se si desidera creare Rust codice utilizzando funzionalità all'avanguardia del linguaggio. Rust

Durante il ciclo di vita di AL2023, le vecchie versioni dei pacchetti non vengono rimosse dai repository. Se è necessaria una Rust toolchain precedente, puoi scegliere di rinunciare alle correzioni

di bug e di sicurezza delle Rust toolchain più recenti e installare una versione precedente dai repository utilizzando gli stessi meccanismi disponibili per qualsiasi RPM.

Se desideri creare il tuo Rust codice AL2023, puoi utilizzare la Rust toolchain inclusa nella AL2023 confezione sapendo che questa toolchain potrebbe andare avanti per tutta la vita di AL2023

AL2023 Funzioni Lambda scritte in Rust

Poiché viene Rust compilato in codice nativo, Lambda Rust tratta come un runtime personalizzato. Puoi usare il `provided.al2023` runtime per distribuire Rust funzioni su AL2023 Lambda.

Per ulteriori informazioni, consulta [Creazione di funzioni Lambda Rust nella Guida](#) per gli AWS Lambda sviluppatori.

Swift nel AL2023

AL2023 attualmente fornisce il [Swift](#) runtime e l'SDK.

In AL2023, l'SwiftSDK e il runtime sono suddivisi in pacchetti separati. Il `swiftlang-lib` pacchetto contiene le librerie di runtime e il `swiftlang` pacchetto contiene l'SDK completo. Ciò consente agli utenti di eseguire i file binari compilati con l'SwiftSDK sugli host di produzione senza richiedere l'installazione dell'SDK completo. Al momento, diverse versioni di non Swift sono compatibili con ABI, quindi durante l'aggiornamento Swift, le applicazioni dovranno essere ricostruite con la nuova versione prima di poter essere distribuite.

Note

Ad esempio Rust, Swift le versioni seguiranno l'ultima versione stabile del Swift progetto. I pacchetti non avranno un namespace e sarà supportata solo una versione alla volta.

Alcune note sull'utilizzo di Swift AL2023

- Interoperabilità C++: il gestore di pacchetti del sistema `clang` potrebbe non funzionare con. Swift Per risolvere questo problema, `swift-clang` aggiungete `PATH` o utilizzate alternative su cui impostare. `clang swift-clang`
- Debug del codice Swift: il sistema LLDB non può eseguire il debug dei sorgenti. Swift Utilizza invece quello LLDB incluso nel Swift pacchetto aggiungendo la directory di Swift installazione alla tua `PATH` o utilizzando alternative su cui impostare. `lldb swift-lldb`

TypeScript nel AL2 2023

Note

Questo documento fornisce le informazioni essenziali sul relativo ambiente di esecuzione TypeScript basato su Node.js. Descrive inoltre un tipico flusso di lavoro di sviluppo e spiega come TypeScript è confezionato in AL2 023 per fornire un ambiente di sviluppo coerente e riproducibile.

[TypeScript](#) (TS) è un linguaggio di programmazione basato su JavaScript (JS) che offre tutte le funzionalità di JS e [lo estende](#) anche con un sistema di tipi. In uno scenario tipico, i programmi scritti in TS vengono prima tradotti nel codice JS e quindi eseguiti come qualsiasi altro normale programma JS da Node.js. Nella terminologia specifica di TS, questo processo di traduzione è chiamato [«compilazione» e viene eseguito da un «compilatore»](#), chiamato tsc. Il compilatore tsc stesso è scritto in JS, quindi per funzionare necessita anche di un ambiente di runtime JS, ad esempio Node.js. A differenza di altri ambienti di runtime JS, Node.js attualmente ha solo un supporto TS sperimentale e leggero. Un supporto TS completo, incluso il controllo dei tipi, richiede ancora l'utilizzo di pacchetti di terze parti, come il [typescript](#). Il modo previsto per ottenere tsc (il compilatore TS) per l'ambiente di Node.js runtime consiste nell'installare il modulo del nodo typescript. Questo può essere fatto usando uno dei gestori di pacchetti, in genere npm. Esistono due modi per installare il compilatore TS usando npm: globalmente e in un progetto. [Il metodo ufficialmente consigliato consiste nell'installare](#) il compilatore TS per progetto, il che garantisce la coerenza e la riproducibilità a lungo termine dei progetti. Tuttavia, l'installazione globale del compilatore TS potrebbe comunque essere utile, perché fornisce la stessa versione per l'intero host e il relativo runtime JS, e quindi per i progetti che non hanno un compilatore TS installato localmente. Questo è esattamente il modo in cui i pacchetti RPM disponibili su Amazon Linux, come `nodejs20-typescript` o `nodejs22-typescript`, installano un compilatore TS: globalmente a livello di sistema e separatamente per ogni versione supportata di Node.js.

Il tsc non dipende direttamente da nessuna versione. Node.js [Il compilatore si aspetta un certo livello di funzionalità di runtime, che sono definite in un file speciale \(tsconfig.json\) tramite opzioni come target e lib](#). I valori di queste opzioni rappresentano una versione dello standard [ECMAScript](#) (ES), che può (o meno) essere supportata dall'ambiente di runtime JS. Diverse versioni di Node.js supportano diverse versioni dello standard ES. Più recente è la versione di Node.js, più alta e completa è la versione standard ES supportata. Se `tsconfig.json` non esiste nella directory principale di un progetto, verrà utilizzato il set predefinito di opzioni di configurazione. [La tabella di compatibilità](#)

[con le diverse versioni Node.js e le funzionalità supportate delle varie versioni standard di ES è disponibile all'indirizzo `node.green`](#). Il `tsc` ha più di 100 diverse opzioni, che possono essere definite in `tsconfig.json`. Il concatenamento della configurazione è supportato anche quando alcune opzioni di configurazione sono definite in un altro file e quindi incluse nel file principale. Questo approccio consente di installare un [Config TS di base](#) compatibile con una determinata versione di Node.js e quindi estenderlo con opzioni specifiche del progetto. Fortunatamente, i Base TS Configs for Node.js sono disponibili come moduli di nodi che possono essere installati in una cartella di progetto utilizzando `npm`. [Ecco il codice sorgente della configurazione per le Node.js versioni 18, 20 e 22.](#)

La progettazione Node.js basata sul runtime presenta un certo punto debole: supporta solo una versione del runtime su un host e richiede la riproducibilità e la coerenza di tutte le dipendenze a livello di progetto. Ciò ha portato al seguente approccio comune di utilizzo TypeScript: il compilatore TS, la configurazione di base TS per la Node.js versione corrente e tutte le dipendenze software vengono installate localmente, all'interno di un progetto. Sebbene si preveda che i moduli di nodo installati a livello globale siano solo strumenti CLI, come `npm`, `tsc`, che è anche uno strumento CLI, viene installato raramente a livello globale. Fortunatamente, le installazioni globali (a livello di sistema) e locali (all'interno di un progetto) di `tsc` possono coesistere senza problemi e possono anche essere versioni diverse che vengono utilizzate indipendentemente. Nota che un `tsc` installato localmente deve essere eseguito utilizzando lo strumento `npx`, che viene installato insieme a `npm`. Pertanto, anche con un compilatore TS di sistema, gli utenti hanno la possibilità di scegliere versioni dei componenti del runtime, ad esempio Node.js (cambiando la versione attiva tramite `alternatives`), un compilatore TS (installandolo localmente o globalmente e cambiando anche la versione attiva tramite `alternatives`) e configurandolo per esigenze specifiche.

Amazon Linux impacchetta un compilatore TS allo stesso modo di altri moduli di nodi installati a livello globale, come `npm`, su una versione base per Node.js versione. I pacchetti e i file binari hanno uno spazio dei nomi e contengono la versione principale di come parte dei Node.js loro nomi. Il nome eseguibile predefinito del compilatore, `tsc`, è gestito in fase di esecuzione dallo strumento `alternatives` e indica la versione attualmente attiva Node.js per la quale è stato installato e da cui verrà eseguito. Questa selezione non dipende dalla versione di runtime corrente. Node.js È possibile avere l'eseguibile del nodo che punta a Node.js 20 e `tsc` configurato per essere interpretato dal 22. Node.js È anche possibile utilizzare i nomi con namespace di un compilatore TS, ad esempio `tsc-{MAJOR_VERSION}` indipendentemente da ciò per cui è configurato il nome `tsc` predefinito.

Alcuni comandi utili per gestire la versione attiva di un compilatore TS

1. Controlla per quali alternative è configurato

```
alternatives --list
```

2. Controlla la configurazione attuale di tsc

```
alternatives --display tsc
```

3. Cambia la versione in modo interattivo tsc

```
alternatives --config tsc
```

4. Passa alla modalità manuale e seleziona una versione specifica

```
alternatives --set tsc /usr/bin/tsc-{MAJOR_VERSION}
```

5. Torna alla modalità di selezione automatica della versione

```
alternatives --auto tsc
```

Un esempio di installazione e utilizzo di diverse versioni di Node e di un compilatore TS sullo stesso sistema:

```
# Check the AL2023 release
$ cat /etc/amazon-linux-release
Amazon Linux release 2023.9.20250929 (Amazon Linux)

# Install a TypeScript compiler for Node.js 20 and 22
# Node.js 20 and 22 will be installed automatically
$ sudo dnf install -qy nodejs20-typescript nodejs22-typescript

# Check what was installed
$ rpm -q nodejs20 nodejs20-typescript nodejs22 nodejs22-typescript
nodejs20-20.19.5-1.amzn2023.0.1.x86_64
nodejs20-typescript-5.9.2-1.amzn2023.0.1.noarch
nodejs22-22.19.0-1.amzn2023.0.1.x86_64
nodejs22-typescript-5.9.2-1.amzn2023.0.1.noarch

# Check the active version of Node.js - it is version 20
$ alternatives --display node
node - status is auto.
```

```
link currently points to /usr/bin/node-20
/usr/bin/node-20 - priority 100
slave npmrc: /usr/lib/nodejs20/lib/node_modules/npm/npmrc
slave npm: /usr/bin/npm-20
slave npx: /usr/bin/npx-20
slave node_modules: /usr/lib/nodejs20/lib/node_modules
/usr/bin/node-22 - priority 100
slave npmrc: /usr/lib/nodejs22/lib/node_modules/npm/npmrc
slave npm: /usr/bin/npm-22
slave npx: /usr/bin/npx-22
slave node_modules: /usr/lib/nodejs22/lib/node_modules
Current 'best' version is /usr/bin/node-20.

# Check the active JS runtime version for TypeScript
# Currently, the tsc compiler will be executed by Node.js 22
$ alternatives --display tsc
tsc - status is auto.
link currently points to /usr/bin/tsc-22
/usr/bin/tsc-22 - priority 100
slave tsserver: /usr/bin/tsserver-22
/usr/bin/tsc-20 - priority 100
slave tsserver: /usr/bin/tsserver-20
Current 'best' version is /usr/bin/tsc-22.

# Check versions printed by executables
$ node -v
v20.19.5

$ tsc -v
Version 5.9.2

# while the node is 20, tsc is executed by node 22 anyway
$ head -1 /usr/bin/tsc
#!/usr/bin/node-22

# However, instead of default executable names, e.g. node or tsc,
# we can use namespaced names to target any installed version
$ node-20 -v
v20.19.5

$ node-22 -v
v22.19.0

$ tsc-20 -v
```

```
Version 5.9.2
```

```
$ tsc-22 -v  
Version 5.9.2
```

```
$ head -1 /usr/bin/tsc-20  
#!/usr/bin/node-20
```

```
$ head -1 /usr/bin/tsc-22  
#!/usr/bin/node-22
```

AL2023 Utenti e gruppi riservati

AL2023 prealloca determinati utenti e gruppi sia durante la fornitura dell'immagine che durante l'installazione di determinati pacchetti. Gli utenti, i gruppi e i relativi UIDs annunci GIDs sono elencati qui per prevenire i conflitti.

Argomenti

- [Elenco di AL2 023 utenti riservati](#)
- [Elenco di AL2 023 gruppi riservati](#)

Elenco di AL2 023 utenti riservati

Nome utente	UID
root	0
bin	1
demone	2
adm	3
lp	4
sincronizzare	5
shutdown	6
fermare	7
posta	8
operatore	11
giochi	12
ftp	14

Nome utente	UID
calamaro	23
denominato	25
postgres	26
mysql	27
nscd	28
nscd	28
utente rpc	29
rpc	32
posta null	47
apache	48
smmsp	51
gatto	53
ldap	55
tss	59
nslcd	65
avahi	70
tcpdump	72
sshd	74
radvd	75
dbus	81

Nome utente	UID
postfix	89
colombaia	97
staffa	156
stapsys	157
stapdev	158
avahi-autoipd	170
impulso	171
art kit	172
sanlock	179
systemd-network	192
sistema-d-resolve	193
uuid	961
stap-server	962
systemd-journal-remote	963
redis6	970
pesigare	971
smtpq	972
smtpd	973
nginx	974
munge	975

Nome utente	UID
memcached	976
sphinx	977
haproxy	978
flatpak	979
eseguire il debug di fod	980
tortora nullo	981
dnsmasq	982
sciolto	983
vongole	984
clamoroso	985
aggiornamento del clam	986
colori	987
odi	988
aws-kinesis-agent-user	989
saslauth	990
agente cw	991
educato	992
ec2-instance-connect	993
chrony	994
systemd-timesync	995

Nome utente	UID
systemd-coredump	996
libstoragemgmt	997
systemd-oom	999
utente ec2	1000
nessuno	65534

Elencato per nome

Nome utente	UID
adm	3
apache	48
avahi	70
avahi-autoipd	170
aws-kinesis-agent-user	989
bin	1
chrony	994
clamoroso	985
vongole	984
aggiornamento del clam	986
colori	987
agente cw	991

Nome utente	UID
demone	2
dbus	81
debug per od	980
dnsmasq	982
colombaia	97
tortora nullo	981
ec2-instance-connect	993
utente ec2	1000
flatpak	979
ftp	14
giochi	12
fermare	7
haproxy	978
saltare	55
libstoragemgmt	997
lp	4
posta	8
mailnull	47
memcached	976
munge	975

Nome utente	UID
mysql	27
denominato	25
nginx	974
nessuno	65534
nscd	28
nscd	28
nslcd	65
od	988
operatore	11
pesigare	971
educato	992
postfix	89
postgres	26
impulso	171
raduno	75
redis6	970
root	0
rpc	32
utente rpc	29
rtkit	172

Nome utente	UID
sanlock	179
saslauth	990
shutdown	6
smmsp	51
smtpd	973
smtpq	972
sphinx	977
calamaro	23
sshd	74
server stap	962
stapdev	158
stafilococco	157
staffa	156
sincronizzare	5
systemd-coredump	996
systemd-journal-remote	963
systemd-network	192
systemd-oom	999
systemd-resolve	193
systemd-timesync	995

Nome utente	UID
tcpdump	72
gatto	53
tss	59
sciolto	983
uid	961

Elenco di AL2 023 gruppi riservati

Group name (Nome gruppo)	GID
root	0
bin	1
demone	2
sys	3
adm	4
tenta	5
disco	6
disco	6
lp	7
mem	8
kmem	9
ruota	10

Group name (Nome gruppo)	GID
cdrom	11
posta	12
posta	12
man	15
dialout	18
floppy	19
giochi	20
slocare	21
utmp	22
calamaro	23
denominato	25
postgres	26
mysql	27
nscd	28
nscd	28
utente rpc	29
rpc	32
registrare	33
utempter	35
kvm	36

Group name (Nome gruppo)	GID
video	39
mailnull	47
apache	48
ftp	50
smmsp	51
gatto	53
serratura	54
ldap	55
tss	59
audio	63
avahi	70
tcpdump	72
sshd	74
radvd	75
saslauth	76
dbus	81
screen	84
wbpriv	88
postfix	89
postrilasciare	90

Group name (Nome gruppo)	GID
colombaia	97
utenti	100
input	104
rendere	105
sgx	106
finto	135
graffetta	156
stafilococco	156
stapsys	157
stafiloide	157
stapdev	158
stapdev	158
avahi-autoipd	170
impulso	171
art kit	172
sanlock	179
systemd-journal	190
systemd-network	192
sistema-d-resolve	193
convocazione	959

Group name (Nome gruppo)	GID
wireshark	960
uuid	961
stap-server	962
systemd-journal-remote	963
gli utenti condividono	964
redis6	965
pesigare	966
smtpq	967
smtpd	968
nginx	969
munge	970
memcached	971
sphinx	972
tracciato	973
haproxy	974
flatpak	975
eseguire il debug di fod	976
dove null	977
dnsmasq	978
sciolto	979

Group name (Nome gruppo)	GID
vongole	980
clamoroso	981
gruppo di virus	982
gruppo di virus	982
gruppo di virus	982
aggiornamento del clam	983
admin di stampa	984
colori	985
odi	986
docker	987
aws-kinesis-agent-user	988
agente	989
impulso-rt	990
accesso a impulsi	991
ec2-instance-connect	993
chrony	994
systemd-timesync	995
systemd-coredump	996
libstoragemgmt	997
ssh_keys	998

Group name (Nome gruppo)	GID
systemd-oom	999
utente ec2	1000
nuovo	1001
educato	920
nessuno	65534

Elencato per nome

Group name (Nome gruppo)	GID
adm	4
apache	48
audio	63
avahi	70
avahi-autoipd	170
aws-kinesis-agent-user	988
bin	1
cdrom	11
chrony	994
clamoroso	981
vongole	980
aggiornamento del clam	983

Group name (Nome gruppo)	GID
colori	985
agente cw	989
demone	2
dbus	81
debug per od	976
dialout	18
disco	6
disco	6
dnsmasq	978
docker	987
colombaia	97
tortora nullo	977
ec2-instance-connect	993
utente ec2	1000
flatpak	975
floppy	19
ftp	50
giochi	20
haproxy	974
input	104

Group name (Nome gruppo)	GID
meme	9
kvm	36
ldap	55
libstoragemgmt	997
serratura	54
lp	7
posta	12
posta	12
mailnull	47
man	15
mem	8
memcached	971
finto	135
munge	970
mysql	27
denominato	25
uno	1001
nginx	969
nessuno	65534
nscd	28

Group name (Nome gruppo)	GID
nscd	28
od	986
pesigano	966
educato	920
postdrop	90
postfix	89
postgres	26
admin di stampa	984
impulso	171
accesso a impulsi	991
impulso-rt	990
radvd	75
redis6	965
rendere	105
root	0
rpc	32
utente rpc	29
rtkit	172
sanlock	179
saslauth	76

Group name (Nome gruppo)	GID
screen	84
sgx	106
sloccare	21
smmsp	51
smtpd	968
smtpq	967
sphinx	972
calamaro	23
ssh_keys	998
sshd	74
server stap	962
stapdev	158
stapdev	158
stafilococco	157
stafiloide	157
staffa	156
stafilococco	156
sys	3
systemd-coredump	996
systemd-journal	190

Group name (Nome gruppo)	GID
systemd-journal-remote	963
systemd-network	192
systemd-oom	999
systemd-resolve	193
systemd-timesync	995
registrare	33
tcpdump	72
gatto	53
tracciato	973
tss	59
tenta	5
sciolto	979
convocazione	959
utenti	100
gli utenti condividono	964
utempter	35
utmp	22
uuuid	961
video	39
gruppo di virus	982

Group name (Nome gruppo)	GID
gruppo di virus	982
gruppo di virus	982
wbpriv	88
ruota	10
wireshark	960

Elenco dei codec disponibili in 023 AL2

AL2023 fornisce una selezione di codec multimediali tramite i suoi archivi standard. Questa pagina fornisce una panoramica dei codec e dei loro casi d'uso tipici.

Important

L'uso e la distribuzione dei codec inclusi in Amazon Linux possono richiedere l'ottenimento di diritti di licenza da terze parti, inclusi proprietari o licenziatari di determinati formati audio e video di terze parti. L'utente è l'unico responsabile dell'ottenimento di tali licenze e del pagamento di eventuali royalty o commissioni necessarie.

Codec	Description
flac	Un codec audio gratuito e open source senza perdita di dati che comprime l'audio senza perdere dati o qualità, comunemente usato per l'archiviazione audio di alta qualità
fdk-aac-free	Un'implementazione open source dello standard AAC (Advanced Audio Codec), che fornisce una compressione audio di alta qualità per MP3 alternative come lo streaming o l'archiviazione di file
webrtc-audio-processing	Una libreria per l'elaborazione audio utilizzata in WebRTC (Web Real-Time Communication), che offre funzionalità come la soppressione del rumore, la cancellazione dell'eco e il controllo del guadagno
opus	Un codec audio altamente versatile ed efficiente e progettato per lo streaming in tempo reale, che offre bassa latenza e supporto per un'ampia gamma di applicazioni audio, tra cui VoIP e streaming musicale

Codec	Description
libsndfile	Una libreria per la lettura e la scrittura di file audio in vari formati (come WAV, AIFF e FLAC), comunemente utilizzata negli strumenti di elaborazione e manipolazione dell'audio
openh264	Un'implementazione open source del codec video H.264. Fornisce supporto per la codifica e la decodifica di video H.264 utilizzati in applicazioni di streaming, conferenze e multimediali.
svt-av1	Un'implementazione open source e ad alte prestazioni del codec video. AV1 Fornisce codifica e decodifica scalabili per AV1 video, ottimizzate per l'elaborazione moderna e CPUs parallela.
dav1d	Un'implementazione open source del decoder AV1 video incentrata su velocità, efficienza e portabilità. Fornisce una AV1 decodifica ad alte prestazioni per un'ampia gamma di piattaforme e applicazioni.
libde265	Un'implementazione open source del decodificatore video H.265/HEVC. Fornisce una decodifica efficiente e portatile dei flussi video HEVC per l'uso in applicazioni e framework multimediali.
libheif	Una libreria open source per leggere e scrivere file di immagine HEIF e AVIF. Fornisce codifica, decodifica e conversione efficienti di immagini e sequenze di immagini utilizzando formati di compressione moderni come HEVC e AV1

Codec	Description
mpg123	Un decoder ad alte prestazioni per flussi audio MPEG, compresi i file. MP3 Fornisce una decodifica audio rapida e precisa per applicazioni di riproduzione, streaming ed elaborazione audio.
x265	L'obiettivo principale di x265 è diventare il miglior encoder H.265/HEVC disponibile ovunque, offrendo la massima efficienza di compressione e le massime prestazioni su un'ampia varietà di piattaforme hardware. Questo pacchetto contiene l'encoder a riga di comando.

Sicurezza e conformità in Amazon Linux 2023

Important

Se desideri segnalare una vulnerabilità o hai un problema di sicurezza relativo ai servizi AWS cloud o ai progetti open source, contatta AWS Security utilizzando la pagina [Vulnerability Reporting](#)

La sicurezza del cloud AWS è la massima priorità. In qualità di AWS cliente, puoi beneficiare di un data center e di un'architettura di rete progettati per soddisfare i requisiti delle organizzazioni più sensibili alla sicurezza.

La sicurezza è una responsabilità condivisa tra AWS e te. Il [modello di responsabilità condivisa](#) descrive questo aspetto come sicurezza del cloud e sicurezza nel cloud:

- Sicurezza del cloud: AWS è responsabile della protezione dell'infrastruttura che gestisce AWS i servizi nel AWS cloud. AWS ti fornisce anche servizi che puoi utilizzare in modo sicuro. I revisori esterni testano e verificano regolarmente l'efficacia della nostra sicurezza nell'ambito dei [AWS Programmi di AWS conformità dei Programmi di conformità](#) dei di . Per maggiori informazioni sui programmi di conformità applicabili AL2023, consulta la sezione [AWS Servizi rientranti nell'ambito del programma di conformitàAWS](#) .
- Sicurezza nel cloud: la tua responsabilità è determinata dal servizio AWS che utilizzi. Inoltre, sei responsabile anche di altri fattori, tra cui la riservatezza dei dati, i requisiti dell'azienda e le leggi e le normative applicabili.

Argomenti

- [Avvisi di sicurezza di Amazon Linux per AL2023](#)
- [Elenco degli avvisi applicabili](#)
- [Applicazione degli aggiornamenti di sicurezza in loco](#)
- [Impostazione SELinux delle modalità per AL2 023](#)
- [Abilita la modalità FIPS su 023 AL2](#)
- [Abilita la modalità FIPS in un contenitore AL2 023](#)
- [Scambia i provider FIPS OpenSSL su 023 AL2](#)

- [AL2023 Indurimento del kernel](#)
- [Accesso ai metadati del repository AL2023](#)
- [Avvio sicuro UEFI attivo AL2023](#)

Avvisi di sicurezza di Amazon Linux per AL2023

Anche se ci impegniamo a fondo per rendere sicuro Amazon Linux, talvolta si verificheranno problemi di sicurezza che devono essere risolti. Quando è disponibile una correzione, viene emesso un avviso. La sede principale in cui pubblichiamo gli avvisi è Amazon Linux Security Center (ALAS). Per ulteriori informazioni, consulta la pagina [Amazon Linux Security Center](#).

Important

[Se desideri segnalare una vulnerabilità o hai un problema di sicurezza relativo ai servizi AWS cloud o ai progetti open source, contatta AWS Security utilizzando la pagina Vulnerability Reporting](#)

Le informazioni sui problemi e sugli aggiornamenti pertinenti AL2023 sono pubblicate dal team di Amazon Linux in diverse località. È comune che gli strumenti di sicurezza recuperino informazioni da queste fonti principali e ti presentino i risultati. Pertanto, potresti non interagire direttamente con le fonti primarie pubblicate da Amazon Linux, ma con l'interfaccia fornita dai tuoi strumenti preferiti, come Amazon [Inspector](#).

Annunci sull'Amazon Linux Security Center

Gli annunci di Amazon Linux vengono forniti per articoli che non rientrano in un avviso. Questa sezione contiene annunci riguardanti ALAS stesso, oltre a informazioni che non rientrano in un avviso. Per ulteriori informazioni, consulta gli [annunci di Amazon Linux Security Center \(ALAS\)](#).

Ad esempio, l'annuncio [2021-001 - Amazon Linux Hotpatch per Apache Log4j rientra in un annuncio](#) piuttosto che in un avviso. In questo annuncio, Amazon Linux ha aggiunto un pacchetto per aiutare i clienti a mitigare un problema di sicurezza in software che non faceva parte di Amazon Linux.

L'[Amazon Linux Security Center CVE Explorer](#) è stato annunciato anche negli annunci ALAS. [Per ulteriori informazioni, consulta Nuovo sito web per CVEs](#)

Domande frequenti su Amazon Linux Security Center

Per le risposte ad alcune domande frequenti su ALAS e sulle modalità di valutazione di Amazon Linux CVEs, consulta [Domande frequenti su Amazon Linux Security Center \(ALAS\)](#) (). FAQs

Avvisi ALAS

Un Amazon Linux Advisory contiene informazioni importanti relative agli utenti di Amazon Linux, in genere informazioni sugli aggiornamenti di sicurezza. L'[Amazon Linux Security Center](#) è il luogo in cui gli avvisi sono visibili sul Web. Le informazioni di avviso fanno anche parte dei metadati del repository dei pacchetti RPM.

Avvisi e archivi RPM

Un repository di pacchetti Amazon Linux 2023 può contenere metadati che descrivono zero o più aggiornamenti. Il `dnf updateinfo` comando prende il nome dal nome del file di metadati del repository che contiene queste informazioni, `updateinfo.xml`. Sebbene il comando abbia un nome `updateinfo` e il file di metadati faccia riferimento a `update`, tutti si riferiscono agli aggiornamenti dei pacchetti che fanno parte di un avviso.

Gli avvisi di Amazon Linux sono pubblicati sul sito Web di [Amazon Linux Security Center](#), insieme alle informazioni presenti nei metadati del repository RPM a cui fa riferimento il gestore di `dnf` pacchetti. I metadati del sito Web e del repository alla fine sono coerenti e potrebbero esserci incongruenze nelle informazioni sul sito Web e nei metadati del repository. Ciò si verifica in genere quando AL2023 è in corso il rilascio di una nuova versione di o è stato aggiornato un avviso dopo la versione più recente. AL2023

Sebbene sia normale che venga emesso un nuovo avviso insieme all'aggiornamento del pacchetto che risolve il problema, non è sempre così. È possibile creare un avviso per un nuovo problema risolto in pacchetti già rilasciati. Un avviso esistente può anche essere aggiornato con nuovi CVEs che vengono risolti dall'aggiornamento esistente.

La [Aggiornamenti deterministici tramite repository con versioni su AL2023](#) funzionalità di Amazon Linux 2023 significa che l'archivio RPM per una particolare AL2023 versione contiene un'istantanea dei metadati del repository RPM a partire da quella versione. Ciò include i metadati che descrivono gli aggiornamenti di sicurezza. L'archivio RPM per una particolare AL2023 versione non viene aggiornato dopo il rilascio. Gli avvisi di sicurezza nuovi o aggiornati non saranno visibili quando si consulta una versione precedente degli AL2023 archivi RPM. Fate riferimento alla [Elenco degli avvisi](#)

[applicabili](#) sezione su come usare il gestore di `dnf` pacchetti per esaminare la versione del `latest` repository o una versione specifica. AL2023

Avviso IDs

Ogni avviso è indicato da un `id`. Attualmente è una stranezza di Amazon Linux dove il sito Web di [Amazon Linux Security Center](#) elenca un avviso come [ALAS-2024-581](#), mentre il gestore di `dnf` pacchetti elenca tale avviso come avente l'ID `-2024-581`. ALAS2023 Quando è necessario utilizzare l'ID del gestore di pacchetti per fare riferimento a un [Applicazione degli aggiornamenti di sicurezza in loco](#) avviso specifico.

Per Amazon Linux, ogni versione principale del sistema operativo ha il proprio namespace di Advisory IDs. Non si devono fare ipotesi sul formato di Amazon Linux Advisory IDs. Storicamente, Amazon Linux Advisory IDs ha seguito lo schema di `NAMESPACE-YEAR-NUMBER`. L'intera gamma di valori possibili per `namespace` è definita, ma include `ALAS`, `ALASCORRETTO8`, `ALAS2023`, `ALAS2ALASPYTHON3.8`, e `ALASUNBOUND-1.17`. `YEAR` è stato l'anno in cui è stato creato l'avviso ed è `NUMBER` stato un numero intero univoco all'interno del namespace.

Sebbene Advisory IDs sia in genere sequenziale e nell'ordine in cui vengono rilasciati gli aggiornamenti, ci sono molte ragioni per cui ciò non è possibile, quindi non si deve dare per scontato.

Considera l'Advisory ID come una stringa opaca che è unica per ogni versione principale di Amazon Linux.

In Amazon Linux 2, ogni Extra si trovava in un repository RPM separato e i metadati di Advisory sono contenuti solo all'interno del repository a cui sono pertinenti. Un avviso per un repository non è applicabile a un altro repository. Sul sito Web di [Amazon Linux Security Center](#), esiste attualmente un elenco di avvisi per ogni versione principale di Amazon Linux e non è suddiviso in elenchi per repository.

Poiché AL2023 non utilizza il meccanismo Extras per impacchettare versioni alternative dei pacchetti, attualmente esistono solo due repository RPM, ognuno dei quali contiene avvisi, il repository `core` e il repository `livepatch`. [Kernel Live Patching attivo AL2023](#) Il repository è per `livepatch`.

Data di rilascio dell'avviso e data di aggiornamento dell'avviso

La data di rilascio dell'avviso per Amazon Linux Advisories indica quando l'aggiornamento di sicurezza è stato reso disponibile pubblicamente per la prima volta nell'archivio RPM. Gli avvisi vengono pubblicati sul sito Web di [Amazon Linux Security Center](#) subito dopo che le correzioni sono state rese disponibili per l'installazione tramite l'archivio RPM.

La data di aggiornamento dell'avviso indica quando sono state aggiunte nuove informazioni a un avviso dopo la sua pubblicazione precedente.

Non si devono fare supposizioni tra il numero di AL2023 versione (ad esempio 2023.6.20241031) e la data di pubblicazione degli avvisi consultivi pubblicati insieme a tale versione.

Tipi di avvisi

I metadati del repository RPM supportano avvisi di diversi tipi. Sebbene Amazon Linux abbia emesso quasi universalmente solo avvisi che sono aggiornamenti di sicurezza, ciò non deve essere dato per scontato. È possibile che vengano emessi avvisi relativi a eventi quali correzioni di bug, miglioramenti e nuovi pacchetti e che l'Avviso venga contrassegnato come contenente quel tipo di aggiornamento.

Severità consultive

Ogni avviso ha la propria gravità in quanto ogni problema viene valutato separatamente. È CVEs possibile affrontare più problemi in un unico avviso e ogni CVE può avere una valutazione diversa, ma l'avviso stesso ha una gravità. Possono esserci più avvisi che si riferiscono a un singolo aggiornamento del pacchetto, quindi possono esserci più livelli di severità per un particolare aggiornamento del pacchetto (uno per avviso).

In ordine decrescente di gravità, Amazon Linux ha utilizzato Critical, Important, Moderate e Low per indicare la gravità di un avviso. Gli avvisi di Amazon Linux possono anche non avere una severità, sebbene ciò sia estremamente raro.

Amazon Linux è una delle distribuzioni Linux basate su RPM che utilizza il termine Moderate, mentre altre distribuzioni Linux basate su RPM utilizzano il termine equivalente Medium. Il gestore di pacchetti Amazon Linux considera entrambi i termini come equivalenti e gli archivi di pacchetti di terze parti possono utilizzare il termine Medium.

Gli avvisi di Amazon Linux possono cambiare la gravità nel tempo man mano che si apprendono ulteriori informazioni sui problemi pertinenti affrontati nell'Advisory.

La severità di un avviso in genere tiene traccia del punteggio CVSS più alto valutato da Amazon Linux per il punteggio a CVEs cui fa riferimento l'avviso. Potrebbero esserci casi in cui questo non è il caso. Un esempio potrebbe essere il caso di un problema risolto per il quale non è stato assegnato un CVE.

Consulta le [domande frequenti su ALAS](#) per ulteriori informazioni su come Amazon Linux utilizza le classificazioni di gravità di Advisory.

Avvisi e pacchetti

Possono esserci molti avvisi per un singolo pacchetto e non tutti i pacchetti avranno mai un avviso pubblicato per essi. È possibile fare riferimento a una particolare versione del pacchetto in più avvisi, ciascuno con la propria gravità e CVEs

È possibile che più avvisi per lo stesso aggiornamento del pacchetto vengano emessi contemporaneamente in una nuova AL2023 versione o in rapida successione.

Come altre distribuzioni Linux, possono esserci uno o più pacchetti binari diversi creati dallo stesso pacchetto sorgente. Ad esempio, [ALAS-2024-698](#) è un avviso elencato nella sezione del sito [Web AL2023 di Amazon Linux Security Center](#) come applicabile al pacchetto `mariadb105`. Questo è il nome del pacchetto sorgente e l'Advisory stesso fa riferimento ai pacchetti binari insieme al pacchetto sorgente. In questo caso, più di una dozzina di pacchetti binari vengono creati a partire da un unico pacchetto `mariadb105` sorgente. Sebbene sia estremamente comune che esista un pacchetto binario con lo stesso nome del pacchetto sorgente, questo non è universale.

Sebbene Amazon Linux Advisories abbia in genere elencato tutti i pacchetti binari creati a partire dal pacchetto sorgente aggiornato, ciò non deve essere dato per scontato. Il formato di metadati del gestore di pacchetti e del repository RPM consente la creazione di avvisi che elencano un sottoinsieme dei pacchetti binari aggiornati.

Un avviso particolare può essere applicato anche solo a una particolare architettura di CPU. Possono esserci pacchetti che non sono stati creati per tutte le architetture o problemi che non riguardano tutte le architetture. Nel caso in cui un pacchetto sia disponibile su tutte le architetture ma un problema si riferisca solo a una, Amazon Linux in genere non ha emesso un avviso che si riferisca solo all'architettura interessata, sebbene ciò non sia da presumere.

A causa della natura delle dipendenze dei pacchetti, è comune che un avviso faccia riferimento a un pacchetto, ma l'installazione di tale aggiornamento richiederà altri aggiornamenti del pacchetto, inclusi i pacchetti non elencati nell'avviso. Il gestore dei `dnf` pacchetti gestirà l'installazione delle dipendenze richieste.

Avvisi e CVEs

Un avviso può riguardare zero o più CVEs avvisi e possono esserci più avvisi che fanno riferimento allo stesso CVE.

Un esempio di quando un Advisory può fare riferimento a zero CVEs è quando un CVE non è ancora (o mai) assegnato al problema.

Un esempio di quando più Advisory possono fare riferimento allo stesso CVE quando (ad esempio) il CVE è applicabile a più pacchetti. Ad esempio, [CVE-2024-21208](#) si applica a Corretto 8, 11, 17 e 21. [Ciascuna di queste versioni di Corretto è un pacchetto separato e per ognuno di questi pacchetti è presente un avviso: ALAS-2024-754 per Corretto 8 AL2023, ALAS-2024-753 per Corretto 11, ALAS-2024-752 per Corretto 17 e ALAS-2024-752 per Corretto 21.](#) Sebbene queste versioni di Corretto abbiano tutte lo stesso elenco di CVEs, ciò non dovrebbe essere dato per scontato.

Un particolare CVE può essere valutato in modo diverso per pacchetti diversi. Ad esempio, se in un avviso si fa riferimento a un particolare CVE con un livello di severità importante, è possibile che venga emesso un altro avviso che faccia riferimento allo stesso CVE con una severità diversa.

I metadati del repository RPM consentono di creare un elenco di riferimenti per ogni avviso. Sebbene Amazon Linux in genere utilizzi solo riferimenti CVEs, il formato dei metadati consente altri tipi di riferimento.

I metadati del repository dei pacchetti RPM faranno riferimento solo se è disponibile una correzione. CVEs La [sezione Esplora del sito Web Amazon Linux Security Center](#) contiene informazioni su CVEs ciò che Amazon Linux ha valutato. Questa valutazione può generare un punteggio di base, una severità e uno status CVSS per varie versioni e pacchetti di Amazon Linux. Lo stato di un CVE per una particolare versione o pacchetto di Amazon Linux può essere Not Affected, Pending Fix o No Fix Planned. Lo stato e la valutazione di CVEs possono cambiare molte volte e in qualsiasi modo prima della pubblicazione di un avviso. Ciò include la rivalutazione dell'applicabilità di un CVE ad Amazon Linux.

L'elenco delle persone a CVEs cui fa riferimento un avviso può cambiare dopo la pubblicazione iniziale di tale avviso.

Testo consultivo

Un avviso conterrà anche un testo che descrive il problema o i problemi che hanno portato alla creazione dell'avviso. È normale che questo testo sia il testo CVE non modificato. Questo testo può fare riferimento a numeri di versione upstream in cui è disponibile una correzione che sono diversi dalla versione del pacchetto a cui Amazon Linux ha applicato una correzione. È normale che Amazon Linux esegua il backport delle correzioni delle versioni upstream più recenti. Nel caso in cui il testo dell'Advisory menzioni una versione upstream diversa dalla versione fornita in una versione Amazon Linux, le versioni del pacchetto Amazon Linux nell'Advisory saranno corrette per Amazon Linux.

È possibile che il testo dell'avviso nei metadati del repository RPM sia testo segnaposto semplicemente facendo riferimento al sito Web di [Amazon Linux Security Center](#) per i dettagli.

Avvisi su Kernel Live Patch

Gli avvisi per le patch live sono unici in quanto si riferiscono a un pacchetto diverso (il kernel Linux) rispetto al pacchetto a cui si riferisce l'Advisory (ad esempio). `kernel-livepatch-6.1.15-28.43`

Un avviso per un [Kernel Live Patch](#) farà riferimento ai problemi (ad esempio CVEs) che il particolare pacchetto Live Patch può risolvere per la versione specifica del kernel a cui si applica il pacchetto live patch.

Ogni patch live è per una versione specifica del kernel. Per applicare una patch live per un CVE, è necessario installare il pacchetto live patch giusto per la versione del kernel e applicare la patch live.

Ad esempio, è possibile applicare una patch live a [CVE-2023-6111](#) per le versioni del kernel e. AL2023 6.1.56-82.125 6.1.59-84.139 6.1.61-85.141 [È stata inoltre rilasciata una nuova versione del kernel con una correzione per questo CVE e contiene un avviso separato.](#) Affinché [CVE-2023-6111](#) possa essere risolto su AL2023 una versione del kernel uguale o successiva a quella specificata da [ALAS2023-2023-461](#) deve essere in esecuzione, oppure una delle versioni del kernel con una patch live per questo CVE deve essere in esecuzione con la livepatch applicabile applicata.

Quando sono disponibili nuove patch live per una versione specifica del kernel che ha già una patch live disponibile, viene rilasciata una nuova versione del pacchetto. `kernel-livepatch-KERNEL_VERSION` Ad esempio, l'[ALASLIVEPATCH-2023-003](#) Advisory è stato rilasciato con il `kernel-livepatch-6.1.15-28.43-1.0-1.amzn2023` pacchetto che conteneva tre patch live per il `6.1.15-28.43` kernel. CVEs Successivamente, insieme al `kernel-livepatch-6.1.15-28.43-1.0-2.amzn2023` pacchetto è stato rilasciato l'[ALASLIVEPATCH-2023-009](#) Advisory, un aggiornamento del precedente pacchetto live patch per il `6.1.15-28.43` kernel contenente patch live per altri tre. CVEs C'erano anche altri problemi relativi agli avvisi di live patch per altre versioni del kernel, con pacchetti contenenti patch live per quelle versioni specifiche del kernel.

Per ulteriori informazioni sul kernel live patching, vedere. [Kernel Live Patching attivo AL2023](#)

Per chiunque sviluppi strumenti relativi agli avvisi di sicurezza, si consiglia inoltre di consultare la [Schema XML per avvisi e updateinfo.xml](#) sezione per ulteriori informazioni.

Schema XML per avvisi e `updateinfo.xml`

Il `updateinfo.xml` file fa parte del formato del repository dei pacchetti. Sono i metadati che il gestore di `dnf` pacchetti analizza per implementare funzionalità come e. [Elenco degli avvisi applicabili](#) [Applicazione degli aggiornamenti di sicurezza in loco](#)

Si consiglia di utilizzare l'API del gestore di `dnf` pacchetti anziché scrivere codice personalizzato per analizzare i formati dei metadati del repository. La versione di `dnf` in AL2023 può analizzare sia i formati del repository che quelli del AL2 repository, AL2023 e quindi l'API può essere utilizzata per esaminare le informazioni di avviso per entrambe le versioni del sistema operativo.

[Il progetto RPM Software Management documenta i formati di metadati RPM nel repository rpm-metadata su GitHub](#)

[Per coloro che sviluppano strumenti per analizzare direttamente i metadati, si consiglia vivamente di prestare particolare attenzione alla documentazione di `updateinfo.xml rpm-metadata`](#). La documentazione copre ciò che è stato visto in giro, il che include molte eccezioni a ciò che si può ragionevolmente interpretare come regola per il formato dei metadati.

[C'è anche una serie crescente di esempi di `updateinfo.xml` file reali nel repository -examples on raw-historical-rpm-repository](#) GitHub

Nel caso in cui qualcosa non sia chiaro nella documentazione, puoi aprire un problema sul GitHub progetto in modo che possiamo rispondere alla domanda e aggiornare la documentazione in modo appropriato. Come progetti Open Source, sono benvenute anche le pull request di aggiornamento della documentazione.

Elenco degli avvisi applicabili

Il gestore dei `dnf` pacchetti ha accesso ai metadati che descrivono quali avvisi sono corretti in quali versioni del pacchetto. Può quindi elencare quali avvisi sono applicabili a un'istanza o a un'immagine del contenitore.

Note

Strumenti come questi [AWS Systems Manager](#) possono utilizzare questa funzionalità per mostrare quali aggiornamenti sono rilevanti per una flotta anziché per una singola istanza.

Quando elenchi gli aggiornamenti, puoi indicare di `dnf` esaminare i metadati di una particolare versione AL2 023 o i metadati dell'ultima versione.

Note

Una volta creata, una versione AL2 023 è immutabile. Pertanto, gli avvisi nuovi o aggiornati su [Amazon Linux Security Center](#) vengono aggiunti solo ai metadati delle nuove versioni di 023. AL2

Ora esamineremo alcuni esempi di come gli avvisi si applicano ad alcune AL2 immagini di container 023. Questi comandi funzionano tutti su ambienti non containerizzati come le istanze. EC2

Listing advisories in a specific version

[In questo esempio vedremo quali avvisi della versione 2023.1.20230628 sono rilevanti in un'immagine del contenitore della versione 2023.0.20230315.](#)

Note

[Questo esempio utilizza le versioni 2023.0.20230315 e 2023.1.20230628, che non sono l'ultima versione della 023. Consulta le note di rilascio 023 per le ultime versioni, che contengono gli ultimi aggiornamenti di sicurezza. AL2 AL2](#)

[In questo esempio inizieremo con un'immagine del contenitore per la versione 2023.0.20230315.](#)

Innanzitutto, recuperiamo l'immagine del contenitore dal registro dei contenitori. La `.0` parte finale indica la versione dell'immagine per una particolare versione; questa versione dell'immagine di solito è zero.

```
$ docker pull public.ecr.aws/amazonlinux/amazonlinux:2023.0.20230315.0
2023.0.20230315.0: Pulling from amazonlinux/amazonlinux
b76f3b09316a: Pull complete
Digest: sha256:94e7183b0739140dbd5b639fb7600f0a2299cec5df8780c26d9cb409da5315a9
Status: Downloaded newer image for public.ecr.aws/amazonlinux/
amazonlinux:2023.0.20230315.0
public.ecr.aws/amazonlinux/amazonlinux:2023.0.20230315.0
```

Ora possiamo generare una shell all'interno del contenitore, dalla quale chiederemo di `dnf` elencare quali avvisi sono rilevanti per i pacchetti installati nel contenitore.

```
$ docker run -it public.ecr.aws/amazonlinux/amazonlinux:2023.0.20230315.0
bash-5.2#
```

Il `dnf updateinfo` comando viene ora utilizzato per visualizzare un riepilogo di quali avvisi nella versione [2023.1.20230628](#) sono rilevanti per i pacchetti installati.

```
$ dnf updateinfo --releasever=2023.1.20230628
Amazon Linux 2023 repository          42 MB/s | 15 MB    00:00
Last metadata expiration check: 0:00:02 ago on Mon Jul 22 20:24:24 2024.
Updates Information Summary: available
  8 Security notice(s)
    1 Important Security notice(s)
    5 Medium Security notice(s)
    2 Low Security notice(s)
```

Per ottenere un elenco degli avvisi, è possibile assegnare l'opzione `a. --list` `dnf updateinfo`

```
$ dnf updateinfo --releasever=2023.1.20230628 --list
Last metadata expiration check: 0:01:22 ago on Mon Jul 22 20:24:24 2024.
ALAS2023-2023-193 Medium/Sec.    curl-minimal-8.0.1-1.amzn2023.x86_64
ALAS2023-2023-225 Medium/Sec.    glib2-2.74.7-688.amzn2023.0.1.x86_64
ALAS2023-2023-195 Low/Sec.      libcap-2.48-2.amzn2023.0.3.x86_64
ALAS2023-2023-193 Medium/Sec.    libcurl-minimal-8.0.1-1.amzn2023.x86_64
ALAS2023-2023-145 Low/Sec.      libgcc-11.3.1-4.amzn2023.0.3.x86_64
ALAS2023-2023-145 Low/Sec.      libgomp-11.3.1-4.amzn2023.0.3.x86_64
ALAS2023-2023-145 Low/Sec.      libstdc++-11.3.1-4.amzn2023.0.3.x86_64
ALAS2023-2023-163 Medium/Sec.    libxml2-2.10.4-1.amzn2023.0.1.x86_64
ALAS2023-2023-220 Important/Sec.  ncurses-base-6.2-4.20200222.amzn2023.0.4.noarch
ALAS2023-2023-220 Important/Sec.  ncurses-libs-6.2-4.20200222.amzn2023.0.4.x86_64
ALAS2023-2023-181 Medium/Sec.    openssl-libs-1:3.0.8-1.amzn2023.0.2.x86_64
ALAS2023-2023-222 Medium/Sec.    openssl-libs-1:3.0.8-1.amzn2023.0.3.x86_64
```

Listing advisories in the latest version

In questo esempio vedremo quali aggiornamenti sono disponibili nella versione AL2 023 se lanciamo un contenitore della latest versione [2023.4.20240319](#). Al momento della stesura, la latest versione è [2023.5.20240708](#), quindi gli aggiornamenti elencati in questo esempio saranno aggiornati a quella versione.

Note

Questo esempio utilizza le versioni [2023.4.20240319](#) e [2023.5.20240708](#), quest'ultima è [la versione più recente al momento](#) della stesura. [Per ulteriori informazioni sulle versioni più recenti, consulta le note di rilascio 023. AL2](#)

In questo esempio inizieremo con un'immagine del contenitore per la versione [2023.4.20240319](#).

Innanzitutto, recuperiamo l'immagine del contenitore dal registro dei contenitori. La `.1` parte finale indica la versione dell'immagine per una particolare versione. Sebbene la versione dell'immagine sia in genere zero, questo esempio utilizza una versione in cui la versione dell'immagine è una.

```
$ docker pull public.ecr.aws/amazonlinux/amazonlinux:2023.4.20240319.1
2023.4.20240319.1: Pulling from amazonlinux/amazonlinux
6de065fda9a2: Pull complete
Digest: sha256:b4838c4cc9211d966b6ea158dacc9eda7433a16ba94436508c2d9f01f7658b4e
Status: Downloaded newer image for public.ecr.aws/amazonlinux/
amazonlinux:2023.4.20240319.1
public.ecr.aws/amazonlinux/amazonlinux:2023.4.20240319.1
```

Ora possiamo generare una shell all'interno del contenitore, dalla quale controlleremo gli aggiornamenti.

```
$ docker run -it public.ecr.aws/amazonlinux/amazonlinux:2023.4.20240319.1
bash-5.2#
```

Il `dnf updateinfo` comando viene ora utilizzato per visualizzare un riepilogo degli avvisi dell'ultima versione relativi ai pacchetti installati. Al momento della stesura, [2023.1.20230628](#) era l'ultima versione.

```
$ dnf --releasever=latest updateinfo
Amazon Linux 2023 repository          76 MB/s | 25 MB    00:00
Last metadata expiration check: 0:00:04 ago on Mon Jul 22 20:59:54 2024.
Updates Information Summary: available
  9 Security notice(s)
    4 Important Security notice(s)
    4 Medium Security notice(s)
    1 Low Security notice(s)
```

Per ottenere un elenco degli avvisi, l'opzione può essere data a. `--list dnf updateinfo`

```
$ dnf updateinfo --releasever=latest --list
```

```
Last metadata expiration check: 0:00:58 ago on Mon Jul 22 20:59:54 2024.  
ALAS2023-2024-581 Low/Sec.      curl-minimal-8.5.0-1.amzn2023.0.3.x86_64  
ALAS2023-2024-596 Medium/Sec.  curl-minimal-8.5.0-1.amzn2023.0.4.x86_64  
ALAS2023-2024-576 Important/Sec. expat-2.5.0-1.amzn2023.0.4.x86_64  
ALAS2023-2024-589 Important/Sec. glibc-2.34-52.amzn2023.0.10.x86_64  
ALAS2023-2024-589 Important/Sec. glibc-common-2.34-52.amzn2023.0.10.x86_64  
ALAS2023-2024-589 Important/Sec. glibc-minimal-langpack-2.34-52.amzn2023.0.10.x86_64  
ALAS2023-2024-586 Medium/Sec.  krb5-libs-1.21-3.amzn2023.0.4.x86_64  
ALAS2023-2024-581 Low/Sec.      libcurl-minimal-8.5.0-1.amzn2023.0.3.x86_64  
ALAS2023-2024-596 Medium/Sec.  libcurl-minimal-8.5.0-1.amzn2023.0.4.x86_64  
ALAS2023-2024-592 Important/Sec. libnghttp2-1.59.0-3.amzn2023.0.1.x86_64  
ALAS2023-2024-640 Medium/Sec.  openssl-libs-1:3.0.8-1.amzn2023.0.12.x86_64  
ALAS2023-2024-605 Medium/Sec.  python3-3.9.16-1.amzn2023.0.7.x86_64  
ALAS2023-2024-616 Important/Sec. python3-3.9.16-1.amzn2023.0.8.x86_64  
ALAS2023-2024-605 Medium/Sec.  python3-libs-3.9.16-1.amzn2023.0.7.x86_64  
ALAS2023-2024-616 Important/Sec. python3-libs-3.9.16-1.amzn2023.0.8.x86_64
```

Applicazione degli aggiornamenti di sicurezza in loco

Per una panoramica sull'applicazione degli aggiornamenti, vedere [Applicazione degli aggiornamenti di sicurezza utilizzando DNF e le versioni del repository](#). L'opzione `--security` consente di limitare `dnf upgrade` gli aggiornamenti dei pacchetti solo a quelli che dispongono di un avviso. Il resto di questa sezione illustrerà come installare solo aggiornamenti di sicurezza specifici.

Note

Si consiglia di applicare tutti gli aggiornamenti disponibili in una nuova versione AL2 023. Scegliere solo gli aggiornamenti di sicurezza o solo gli aggiornamenti specifici dovrebbe essere l'eccezione piuttosto che la regola.

Applicazione degli aggiornamenti menzionati in un avviso

Gli identificatori degli avvisi nella prima colonna dell'output di `dnf upgradeinfo` possono essere utilizzati per applicare gli aggiornamenti ai pacchetti menzionati nell'avviso. Il gestore dei `dnf` pacchetti può essere incaricato di aggiornare i pacchetti dell'avviso agli ultimi disponibili o solo fino

alle versioni menzionate nell'avviso. Se gli aggiornamenti sono già installati, il comando `update` è impossibile.

Per applicare gli aggiornamenti ai pacchetti interessati solo fino alla versione menzionata nell'avviso, utilizzate il `dnf upgrade-minimal` comando mentre utilizzate l'`--advisory` opzione per specificare l'avviso. [L'esempio seguente è `dnf upgrade-minimal` in esecuzione in un contenitore della versione AL2 023 2023.0.20230315.](#)

```
$ dnf upgrade-minimal -y --releasever=2023.1.20230628 --advisory ALAS2023-2023-193
Amazon Linux 2023 repository                46 MB/s | 15 MB    00:00
Last metadata expiration check: 0:00:03 ago on Mon Jul 22 20:36:13 2024.
Dependencies resolved.
=====
Package                Arch      Version                Repository            Size
=====
Upgrading:
curl-minimal           x86_64    8.0.1-1.amzn2023      amazonlinux           150 k
libcurl-minimal        x86_64    8.0.1-1.amzn2023      amazonlinux           249 k

Transaction Summary
=====
Upgrade 2 Packages

Total download size: 399 k
Downloading Packages:
(1/2): curl-minimal-8.0.1-1.amzn2023.x86_64.rpm 2.7 MB/s | 150 kB    00:00
(2/2): libcurl-minimal-8.0.1-1.amzn2023.x86_64. 3.8 MB/s | 249 kB    00:00
-----
Total                2.5 MB/s | 399 kB    00:00
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing           :                               1/1
  Upgrading           : libcurl-minimal-8.0.1-1.amzn2023.x86_64 1/4
  Upgrading           : curl-minimal-8.0.1-1.amzn2023.x86_64    2/4
  Cleanup            : curl-minimal-7.88.1-1.amzn2023.0.1.x86_64 3/4
  Cleanup            : libcurl-minimal-7.88.1-1.amzn2023.0.1.x86_64 4/4
  Running scriptlet: libcurl-minimal-7.88.1-1.amzn2023.0.1.x86_64 4/4
  Verifying          : libcurl-minimal-8.0.1-1.amzn2023.x86_64 1/4
  Verifying          : libcurl-minimal-7.88.1-1.amzn2023.0.1.x86_64 2/4
  Verifying          : curl-minimal-8.0.1-1.amzn2023.x86_64    3/4
```

```

Verifying      : curl-minimal-7.88.1-1.amzn2023.0.1.x86_64      4/4

Upgraded:
  curl-minimal-8.0.1-1.amzn2023.x86_64  libcurl-minimal-8.0.1-1.amzn2023.x86_64

Complete!

```

Le stesse versioni del pacchetto vengono aggiornate anche se `--releasever=latest` vengono utilizzate come richiesta per eseguire l'aggiornamento minimo richiesto `dnf` per rispondere all'avviso.

L'uso del `dnf upgrade` comando normale con l'`--advisory` opzione aggiornerà i pacchetti pertinenti menzionati nell'avviso alla versione più recente disponibile, che potrebbe essere più recente della versione menzionata nell'avviso.

Note

A meno che il `system-release` pacchetto non venga aggiornato, la versione dei repository AL2 023 su cui `dnf` è bloccato non cambia.

Warning

Quando si installano aggiornamenti da una versione diversa di AL2 023 senza modificare la versione del repository su cui `dnf` è bloccato, è necessario prestare attenzione alle successive operazioni di modifica. `dnf` Ad esempio, quando si installano o si aggiornano i pacchetti, poiché le dipendenze dei pacchetti potrebbero essere cambiate nella versione più recente, la versione precedente utilizzata potrebbe non essere in grado di soddisfare queste nuove dipendenze.

[L'esempio seguente viene eseguito in un contenitore della versione AL2 023 2023.0.20230315 che fa riferimento all'ultima versione di 023, la cui data di AL2 scrittura era 2023.5.20240708.](#) Nota che entrambe le versioni a cui `curl` viene effettuato l'aggiornamento sono più recenti della versione a cui è stato aggiornato, ma che questa versione più recente introduce nuove dipendenze. `update-minimal`

```

$ dnf upgrade -y --releasever=latest --advisory ALAS2023-2023-193
Amazon Linux 2023 repository      80 MB/s | 25 MB      00:00
Last metadata expiration check: 0:00:04 ago on Mon Jul 22 20:48:38 2024.

```

Dependencies resolved.

```

=====
Package                Arch      Version                               Repository      Size
=====
Upgrading:
curl-minimal           x86_64   8.5.0-1.amzn2023.0.4                 amazonlinux     160 k
libcurl-minimal       x86_64   8.5.0-1.amzn2023.0.4                 amazonlinux     275 k
libnghttp2             x86_64   1.59.0-3.amzn2023.0.1                 amazonlinux     79 k
Installing dependencies:
libpsl                 x86_64   0.21.1-3.amzn2023.0.2                 amazonlinux     61 k
publicsuffix-list-dafsa noarch   20240212-61.amzn2023                  amazonlinux     59 k

```

Transaction Summary

```

=====
Install  2 Packages
Upgrade  3 Packages

```

Total download size: 634 k

Downloading Packages:

```

(1/5): publicsuffix-list-dafsa-20240212-61.amzn 1.1 MB/s | 59 kB      00:00
(2/5): curl-minimal-8.5.0-1.amzn2023.0.4.x86_64 2.6 MB/s | 160 kB     00:00
(3/5): libpsl-0.21.1-3.amzn2023.0.2.x86_64.rpm 949 kB/s | 61 kB      00:00
(4/5): libnghttp2-1.59.0-3.amzn2023.0.1.x86_64. 3.7 MB/s | 79 kB      00:00
(5/5): libcurl-minimal-8.5.0-1.amzn2023.0.4.x86 6.7 MB/s | 275 kB     00:00

```

```

-----
Total                                3.5 MB/s | 634 kB     00:00

```

Running transaction check

Transaction check succeeded.

Running transaction test

Transaction test succeeded.

Running transaction

```

Preparing      :                               1/1
Upgrading      : libnghttp2-1.59.0-3.amzn2023.0.1.x86_64 1/8
Installing     : publicsuffix-list-dafsa-20240212-61.amzn2023.noarch 2/8
Installing     : libpsl-0.21.1-3.amzn2023.0.2.x86_64 3/8
Upgrading      : libcurl-minimal-8.5.0-1.amzn2023.0.4.x86_64 4/8
Upgrading      : curl-minimal-8.5.0-1.amzn2023.0.4.x86_64 5/8
Cleanup        : curl-minimal-7.88.1-1.amzn2023.0.1.x86_64 6/8
Cleanup        : libcurl-minimal-7.88.1-1.amzn2023.0.1.x86_64 7/8
Cleanup        : libnghttp2-1.51.0-1.amzn2023.x86_64 8/8
Running scriptlet: libnghttp2-1.51.0-1.amzn2023.x86_64 8/8
Verifying      : libpsl-0.21.1-3.amzn2023.0.2.x86_64 1/8
Verifying      : publicsuffix-list-dafsa-20240212-61.amzn2023.noarch 2/8
Verifying      : curl-minimal-8.5.0-1.amzn2023.0.4.x86_64 3/8

```

```
Verifying      : curl-minimal-7.88.1-1.amzn2023.0.1.x86_64      4/8
Verifying      : libcurl-minimal-8.5.0-1.amzn2023.0.4.x86_64    5/8
Verifying      : libcurl-minimal-7.88.1-1.amzn2023.0.1.x86_64  6/8
Verifying      : libnghttp2-1.59.0-3.amzn2023.0.1.x86_64      7/8
Verifying      : libnghttp2-1.51.0-1.amzn2023.x86_64          8/8
```

Upgraded:

```
curl-minimal-8.5.0-1.amzn2023.0.4.x86_64
libcurl-minimal-8.5.0-1.amzn2023.0.4.x86_64
libnghttp2-1.59.0-3.amzn2023.0.1.x86_64
```

Installed:

```
libpsl-0.21.1-3.amzn2023.0.2.x86_64
publicsuffix-list-dafsa-20240212-61.amzn2023.noarch
```

Complete!

Impostazione SELinux delle modalità per AL2 023

Per impostazione predefinita, Security Enhanced Linux (SELinux) è enabled impostata sulla permissive modalità 023. AL2 In modalità permissiva, i dinieghi di autorizzazione vengono registrati ma non applicati. SELinux è una raccolta di funzionalità e utilità del kernel per fornire un'architettura MAC (Mandatory Access Control) solida, flessibile e obbligatoria ai principali sottosistemi del kernel.

SELinux fornisce un meccanismo avanzato per imporre la separazione delle informazioni in base ai requisiti di riservatezza e integrità. Questa separazione delle informazioni riduce le minacce di manomissione e aggiramento dei meccanismi di sicurezza delle applicazioni. Inoltre limita i danni che possono essere causati da applicazioni dannose o difettose.

SELinux include una serie di esempi di file di configurazione delle politiche di sicurezza progettati per soddisfare gli obiettivi di sicurezza quotidiani.

Per ulteriori informazioni su SELinux caratteristiche e funzionalità, consulta «[SELinux Notebook and Policy Languages](#)».

Argomenti

- [SELinux Stato e modalità predefiniti per AL2 023](#)
- [Passaggio alla modalità enforcing](#)
- [Opzione da disabilitare SELinux per AL2 023](#)

SELinux Stato e modalità predefiniti per AL2 023

Per AL2 023, per impostazione SELinux predefinita è `enabled` e impostato su `mode.permissive`. In modalità `permissive`, le negazioni di autorizzazione vengono registrate ma non applicate.

I `sestatus` comandi `getenforce` or indicano lo SELinux stato, la politica e la modalità correnti.

Con lo stato predefinito impostato su `enabled` e `permissive`, il comando `getenforce` restituisce `permissive`.

Il `sestatus` comando restituisce lo SELinux stato e la SELinux politica corrente, come illustrato nell'esempio seguente:

```
$ sestatus
SELinux status:                enabled
SELinuxfs mount:              /sys/fs/selinux
SELinux root directory:       /etc/selinux
Loaded policy name:           targeted
Current mode:                  permissive
Mode from config file:        permissive
Policy MLS status:           enabled
Policy deny_unknown status:   allowed
Memory protection checking:   actual (secure)
Max kernel policy version:    33
```

Quando si esegue SELinux in `permissive` modalità, gli utenti potrebbero etichettare i file in modo errato. Quando esegui SELinux nello `disabled` stato, i file non sono etichettati. Sia i file errati che quelli senza etichetta possono causare problemi quando passi alla modalità `enforcing`.

SELinux rietichetta automaticamente i file per evitare questo problema. SELinux previene i problemi di etichettatura grazie alla rietichettatura automatica quando si modifica lo stato in `enabled`

Passaggio alla modalità `enforcing`

Quando corri SELinux in `enforcing` modalità, il SELinux l'utilità è `enforcing` la politica configurata. SELinux regola le funzionalità di determinate applicazioni consentendo o negando l'accesso in base alle regole della politica.

Per trovare la corrente SELinux modalità, esegui il `getenforce` comando.

```
getenforce
```

```
Permissive
```

Modifica del file di configurazione per abilitare la modalità **enforcing**

Per cambiare la modalità in `enforcing`, utilizzare la procedura seguente.

1. Modifica il file `/etc/selinux/config` per passare alla modalità `enforcing`. L'`SELINUX` impostazione dovrebbe essere simile all'esempio seguente.

```
SELINUX=enforcing
```

2. Riavvia il sistema per completare il passaggio alla modalità `enforcing`.

```
$ sudo reboot
```

All'avvio successivo, SELinux riassegna tutti i file e le cartelle del sistema. SELinux aggiunge anche il SELinux contesto per i file e le directory che sono stati creati quando SELinux era `disabled`.

Dopo il passaggio alla `enforcing` modalità, SELinux potrebbe negare alcune azioni perché errate o mancanti SELinux regole politiche. È possibile visualizzare le azioni che SELinux nega con il seguente comando.

```
$ sudo ausearch -m AVC,USER_AVC,SELINUX_ERR,USER_SELINUX_ERR -ts recent
```

Utilizzo cloud-init per abilitare la modalità **enforcing**

In alternativa, quando avvii l'istanza, passa la seguente `cloud-config` come dati utente per abilitare la modalità `enforcing`.

```
#cloud-config
selinux:
  mode: enforcing
```

Per impostazione predefinita, questa impostazione causa il riavvio dell'istanza. Per una maggiore stabilità, consigliamo di riavviare l'istanza. Tuttavia, se preferisci, puoi saltare il riavvio fornendo la seguente `cloud-config`.

```
#cloud-config
```

```
selinux:  
  mode: enforcing  
  selinux_no_reboot: 1
```

Opzione da disabilitare SELinux per AL2 023

Quando si disattiva SELinux, SELinux la policy non viene caricata o applicata e i messaggi di Access Vector Cache (AVC) non vengono registrati. Perdi tutti i vantaggi della corsa SELinux.

Invece di disabilitare SELinux, si consiglia di utilizzare la **permissive** modalità. L'esecuzione in **permissive** modalità costa solo un po' di più rispetto alla disattivazione SELinux completamente. La transizione da una **permissive enforcing** modalità all'altra richiede una regolazione della configurazione molto inferiore rispetto al ritorno alla **enforcing** modalità dopo la disabilitazione SELinux. È possibile etichettare i file e il sistema può tenere traccia e registrare le azioni che la politica attiva potrebbe aver negato.

Modifica SELinux alla **permissive** modalità

Quando corri SELinux in **permissive** modalità, SELinux la politica non viene applicata. In **permissive** modalità, SELinux registra i messaggi AVC ma non nega le operazioni. È possibile utilizzare questi messaggi AVC per la risoluzione dei problemi, il debug e SELinux miglioramenti delle politiche.

Cambiare SELinux alla modalità permissiva, utilizzare i seguenti passaggi.

1. Modifica il file `/etc/selinux/config` per passare alla modalità **permissive**. Il `SELINUX` valore dovrebbe essere simile all'esempio seguente.

```
SELINUX=permissive
```

2. Riavvia il sistema per completare il passaggio alla modalità **permissive**.

```
sudo reboot
```

Disabilita SELinux

Quando si disattiva SELinux, SELinux la politica non viene caricata o applicata e i messaggi AVC non vengono registrati. Perdi tutti i vantaggi della corsa SELinux.

Per disabilitare SELinux, attenersi alla seguente procedura.

1. Assicurarsi che il `grubby` pacchetto sia installato.

```
rpm -q grubby
grubby-version
```

2. Configura il bootloader per aggiungere `selinux=0` alla riga di comando del kernel.

```
sudo grubby --update-kernel ALL --args selinux=0
```

3. Riavvia il sistema.

```
sudo reboot
```

4. Esegui il `getenforce` comando per confermarlo SELinux è Disabled.

```
$ getenforce
Disabled
```

Per ulteriori informazioni sull' SELinux, vedi il [SELinux Notebook](#) e [SELinux configurazione](#).

Abilita la modalità FIPS su 023 AL2

Questa sezione spiega come abilitare gli standard federali di elaborazione delle informazioni (FIPS) su 023. AL2 Per ulteriori informazioni sul FIPS, consulta:

- [Federal Information Processing Standard \(FIPS\)](#)
- [Conformità FAQs: standard federali per l'elaborazione delle informazioni](#)

Note

Questa sezione illustra come abilitare FIPS in modalità AL2 023, non copre lo stato di certificazione dei moduli crittografici AL2 023.

Prerequisiti

- Un' EC2 istanza Amazon AL2 023 (AL2023.2 o superiore) esistente con accesso a Internet per scaricare i pacchetti richiesti. Per ulteriori informazioni sul lancio di un' EC2 istanza AL2 Amazon 023, consulta [Avvio di AL2023 tramite la console Amazon EC2](#)
- Devi connetterti alla tua EC2 istanza Amazon tramite SSH o AWS Systems Manager. Per ulteriori informazioni, consulta [AL2023 Connessione alle istanze](#).

Important

ED25519 Le chiavi utente SSH non sono supportate in modalità FIPS. Se hai avviato l' EC2 istanza Amazon utilizzando una coppia di chiavi ED25519 SSH, devi generare nuove chiavi utilizzando un altro algoritmo (come RSA) o potresti perdere l'accesso all'istanza dopo aver abilitato la modalità FIPS. Per ulteriori informazioni, consulta [Create key pair](#) nella Amazon EC2 User Guide.

Abilitazione della modalità FIPS

1. Connect alla propria istanza AL2 023 utilizzando SSH o. AWS Systems Manager
2. Verifica che il sistema sia aggiornato. Per ulteriori informazioni, consulta [Gestisci gli aggiornamenti dei pacchetti e del sistema operativo in AL2023](#).
3. Assicurati che le crypto-policies utilità siano installate e. up-to-date

```
sudo dnf -y install crypto-policies crypto-policies-scripts
```

4. Abilita la modalità FIPS eseguendo il seguente comando. [Ciò abiliterà la modalità FIPS a livello di sistema per i moduli elencati nelle domande frequenti 023 AL2](#)

```
sudo fips-mode-setup --enable
```

5. Riavvia l'istanza utilizzando il comando seguente.

```
sudo reboot
```

6. Per verificare che la modalità FIPS sia abilitata, riconnettiti all'istanza ed esegui il comando seguente.

```
sudo fips-mode-setup --check
```

Il seguente esempio di output mostra che la modalità FIPS è abilitata:

```
FIPS mode is enabled.
```

Abilita la modalità FIPS in un contenitore AL2 023

Questa sezione spiega come abilitare gli standard federali di elaborazione delle informazioni (FIPS) in un contenitore AL2 023. Per ulteriori informazioni sul FIPS, consulta:

- [Federal Information Processing Standard \(FIPS\)](#)
- [Conformità FAQs: standard federali per l'elaborazione delle informazioni](#)

Note

Questa sezione illustra come abilitare FIPS modalità in un contenitore AL2 023. Non copre lo stato di certificazione dei moduli crittografici AL2 023.

Prerequisiti

- Un' EC2 istanza Amazon AL2 023 (AL2023.2 o superiore) esistente con accesso a Internet per scaricare i pacchetti richiesti. Per ulteriori informazioni sul lancio di un' EC2 istanza AL2 Amazon 023, consulta [Avvio di AL2023 tramite la console Amazon EC2](#)
- Devi connetterti alla tua EC2 istanza Amazon tramite SSH o AWS Systems Manager. Per ulteriori informazioni, consulta [AL2023 Connessione alle istanze](#).

Important

Il `fips-mode-setup` comando non funzionerà correttamente dall'interno del contenitore. Leggi i passaggi seguenti per configurare correttamente la modalità FIPS in un contenitore AL2 023.

Abilita la modalità FIPS in un contenitore 023 AL2

1. La modalità FIPS deve essere prima abilitata sull'host del contenitore AL2 023. Segui le istruzioni riportate [Abilita la modalità FIPS su 023 AL2](#) per abilitare la modalità FIPS sull'host.
2. Connect all'istanza host del contenitore AL2 023 utilizzando SSH o. AWS Systems Manager
3. La modalità FIPS verrà abilitata automaticamente in un contenitore AL2 023 se l'host AL2 023 è in modalità FIPS ed `/proc/sys/crypto/fips_enabled` è accessibile dall'interno del contenitore. Se il contenuto di `/proc/sys/crypto/fips_enabled` è, `0` allora FIPS non è abilitato e il valore di `1` indica che la modalità FIPS è abilitata.

È possibile verificare che FIPS sia abilitato eseguendo il comando seguente sia sull'host AL2 023 che sul contenitore:

```
cat /proc/sys/crypto/fips_enabled
```

4. Successivamente, abilita le crypto-policies FIPS all'interno del contenitore. Esistono diversi modi per eseguire questa operazione, descritti nelle opzioni seguenti. Utilizzate l'opzione più adatta al vostro ambiente.

- a. Abilita manualmente le crypto-policies FIPS all'interno del contenitore utilizzando il comando:
`update-crypto-policies`

```
# Run these commands inside the container
dnf install -y crypto-policies-scripts
update-crypto-policies --set FIPS
```

- b. Crea bind mount all'interno del contenitore AL2 023 (è simile a come podman funziona in altre distribuzioni):

```
# Run these commands inside the container
mount --bind /usr/share/crypto-policies/back-ends/FIPS /etc/crypto-policies/back-ends
echo "FIPS" > /usr/share/crypto-policies/default-fips-config
mount --bind /usr/share/crypto-policies/default-fips-config /etc/crypto-policies/config
```

- c. È anche possibile creare un bind mount in modo che il contenitore AL2 023 corrisponda alle crypto-policies dell' AL2host 023. Quanto segue è fornito solo come esempio. Questa configurazione potrebbe causare problemi in caso di differenze incompatibili nelle crypto-policies e nelle versioni dei pacchetti tra il contenitore e l'host:

```
sudo docker pull amazonlinux:2023
sudo docker run --mount type=bind,readonly,src=/etc/crypto-policies,dst=/etc/
crypto-policies -it amazonlinux:2023
```

5. Dopo aver eseguito i passaggi precedenti, puoi verificare nuovamente che FIPS sia abilitato nel contenitore con i seguenti comandi:

```
$ cat /etc/crypto-policies/config
FIPS

$ cat /proc/sys/crypto/fips_enabled
1
```

Scambia i provider FIPS OpenSSL su 023 AL2

Questa sezione spiega come passare dai provider FIPS latest certified OpenSSL su 023. AL2

Per ulteriori informazioni sul FIPS, consulta:

- [Federal Information Processing Standard \(FIPS\)](#)
- [Conformità FAQs: standard federali per l'elaborazione delle informazioni](#)
- [Politica FedRAMP per la selezione e l'uso dei moduli crittografici](#)

Important

A partire dalla versione AL2 023.7, il provider FIPS OpenSSL predefinito è il `openssl-fips-provider-latest` pacchetto, che riceve regolarmente correzioni di bug e aggiornamenti di sicurezza.

Le istruzioni riportate di seguito sono solo per i clienti che desiderano aggiungerlo al pacchetto. `openssl-fips-provider-certified` Questa versione del provider FIPS corrisponderà al checksum del certificato NIST e potrebbe non disporre degli ultimi aggiornamenti.

Consulta le [domande frequenti AL2 023](#) per ulteriori informazioni sui moduli certificati FIPS e sulle versioni dei pacchetti.

Prerequisiti

- Un' EC2 istanza Amazon AL2 023 (AL2023.7 o superiore) esistente con accesso a Internet per scaricare i pacchetti richiesti. Per ulteriori informazioni sul lancio di un' EC2 istanza AL2 Amazon 023, consulta. [Avvio di AL2023 tramite la console Amazon EC2](#)
- Devi connetterti alla tua EC2 istanza Amazon tramite SSH o AWS Systems Manager. Per ulteriori informazioni, consulta [AL2023 Connessione alle istanze](#).
- Per abilitare la modalità FIPS su AL2 023, segui le istruzioni all'indirizzo. [Abilita la modalità FIPS su 023 AL2](#)

Passa da `openssl-fips-provider-latest` a `openssl-fips-provider-certified`

1. Utilizzare `dnf` per cambiare il provider FIPS OpenSSL:

```
sudo dnf -y swap openssl-fips-provider-latest openssl-fips-provider-certified
```

2. Verifica di utilizzare il provider FIPS certificato OpenSSL. Con AL2 023 in modalità FIPS, esegui il seguente comando:

```
openssl list -providers
```

Verrà visualizzato l'output seguente:

```
Providers:
  base
    name: OpenSSL Base Provider
    version: 3.2.2
    status: active
  default
    name: OpenSSL Default Provider
    version: 3.2.2
    status: active
  fips
    name: Amazon Linux 2023 - OpenSSL FIPS Provider
    version: 3.0.8-d694bfa693b76001
    status: active
```

AL2023 Indurimento del kernel

Il kernel Linux 6.1 AL2023 è configurato e costruito con diverse opzioni e funzionalità di rafforzamento.

Opzioni di rafforzamento del kernel (indipendenti dall'architettura)

Opzione CONFIG	AL2023/6. 1/aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/aarch64	AL2023/6. 12/x86_64	AL2023/6. 18/aarch64	AL2023/6. 18/x86_64
<u>CONFIG_ACPI_CUSTOM_METHOD</u>	n	n	N/D	N/D	N/D	N/D
<u>CONFIG_BINFMT_MISC</u>	m	m	m	m	m	m
<u>CONFIG_BUG</u>	y	y	y	y	y	y
<u>CONFIG_BUG_ON_DATA_CORRUPTION</u>	y	y	y	y	y	y
<u>CONFIG_CLANG_I</u>	N/D	N/D	N/D	N/D	N/D	N/D
<u>CONFIG_CLANG_I_PERMISSIVE</u>	N/D	N/D	N/D	N/D	N/D	N/D
<u>CONFIG_COMPAT</u>	y	y	y	y	y	y
<u>CONFIG_COMPAT_BRK</u>	n	n	n	n	n	n

Opzione CONFIG	AL2023/6. 1/aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/aarch64	AL2023/6. 12/x86_64	AL2023/6. 18/aarch64	AL2023/6. 18/x86_64
<u>CONFIG_CO MPAT_VDSO</u>	N/D	n	N/D	n	N/D	n
<u>CONFIG_DE BUG_CREDE NTIALS</u>	n	n	N/D	N/D	N/D	N/D
<u>CONFIG_DE BUG_LIST</u>	y	y	y	y	y	y
<u>CONFIG_DE BUG_NOTIF IERS</u>	n	n	n	n	n	n
<u>CONFIG_DE BUG_SG</u>	n	n	n	n	n	n
<u>CONFIG_DE BUG_VIRTU AL</u>	n	n	n	n	n	n
<u>CONFIG_DE BUG_WX</u>	n	n	n	n	n	n
<u>CONFIG_DE FAULT_MMA P_MIN_ADD R</u>	65536	65536	65536	65536	65536	65536
<u>CONFIG_DE VKMEM</u>	N/D	N/D	N/D	N/D	N/D	N/D
<u>CONFIG_DE VMEM</u>	n	n	n	n	n	n

Opzione CONFIG	AL2023/6. 1/aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/aarch64	AL2023/6. 12/x86_64	AL2023/6. 18/aarch64	AL2023/6. 18/x86_64
<u>CONFIG_EFI_DISABLE_PCI_DMA</u>	n	n	n	n	n	n
<u>CONFIG_FORTIFY_SOURCE</u>	y	y	y	y	y	y
<u>CONFIG_HARDENED_USERCOPY</u>	y	y	y	y	y	y
<u>CONFIG_HARDENED_USERCOPY_FALLBACK</u>	N/D	N/D	N/D	N/D	N/D	N/D
<u>CONFIG_HARDENED_USERCOPY_PAGESPAN</u>	N/D	N/D	N/D	N/D	N/D	N/D
<u>CONFIG_HIBERNATION</u>	y	y	y	y	y	y
<u>CONFIG_HW_RANDOM_TPM</u>	N/D	N/D	N/D	N/D	N/D	N/D
<u>CONFIG_INET_DIAG</u>	m	m	m	m	m	m

Opzione CONFIG	AL2023/6. 1/aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/aarch64	AL2023/6. 12/x86_64	AL2023/6. 18/aarch64	AL2023/6. 18/x86_64
<u>CONFIG_IN IT_ON_ALL OC_DEFAULT T_ON</u>	n	n	n	n	n	n
<u>CONFIG_IN IT_ON_FRE E_DEFAULT _ON</u>	n	n	n	n	n	n
<u>CONFIG_IN IT_STACK_ ALL_ZERO</u>	N/D	N/D	N/D	N/D	N/D	N/D
<u>CONFIG_IO MMU_DEFAU LT_DMA_ST RICT</u>	n	n	n	n	n	n
<u>CONFIG_IO MMU_SUPPO RT</u>	y	y	y	y	y	y
<u>CONFIG_IO _STRICT_D EVMEM</u>	N/D	N/D	N/D	N/D	N/D	N/D
<u>CONFIG_KE XEC</u>	y	y	y	y	y	y
<u>CONFIG_KF ENCE</u>	n	n	n	n	n	n

Opzione CONFIG	AL2023/6. 1/aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/aarch64	AL2023/6. 12/x86_64	AL2023/6. 18/aarch64	AL2023/6. 18/x86_64
<u>CONFIG_LD ISC_AUTOL OAD</u>	n	n	n	n	n	n
<u>CONFIG_LEGACY_PTYS</u>	n	n	n	n	n	n
<u>CONFIG_LOCK_DOWN_KERNEL_FORCE_CONFIDENTIALITY</u>	n	n	n	n	n	n
<u>CONFIG_MODULES</u>	y	y	y	y	y	y
<u>CONFIG_MODULE_SIG</u>	y	y	y	y	y	y
<u>CONFIG_MODULE_SIG_ALL</u>	y	y	y	y	y	y
<u>CONFIG_MODULE_SIG_FORCE</u>	n	n	n	n	n	n
<u>CONFIG_MODULE_SIG_HASH</u>	sha512	sha512	sha512	sha512	sha512	sha512

Opzione CONFIG	AL2023/6. 1/aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/aarch64	AL2023/6. 12/x86_64	AL2023/6. 18/aarch64	AL2023/6. 18/x86_64
<u>CONFIG_MODULE_SIG_KEY</u>	certs/ sig ning_key. pem	certs/ sig ning_key. pem	certs/ sig ning_key. pem	certs/ sig ning_key. pem	certs/ sig ning_key. pem	certs/ sig ning_key. pem
<u>CONFIG_MODULE_SIG_SHA512</u>	y	y	y	y	y	y
<u>CONFIG_PACKAGE_POISONING</u>	n	n	n	n	n	n
<u>CONFIG_PACKAGE_POISONING_NO_SANITY</u>	N/D	N/D	N/D	N/D	N/D	N/D
<u>CONFIG_PACKAGE_POISONING_ZERO</u>	N/D	N/D	N/D	N/D	N/D	N/D
<u>CONFIG_PACKAGE_ON_OOPS</u>	y	y	y	y	y	y
<u>CONFIG_PACKAGE_TIMEOUT</u>	0	0	0	0	0	0
<u>CONFIG_PROC_KCORE</u>	y	y	y	y	y	y

Opzione CONFIG	AL2023/6. 1/aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/aarch64	AL2023/6. 12/x86_64	AL2023/6. 18/aarch64	AL2023/6. 18/x86_64
CONFIG_RA NDOMIZE_K STACK_OFF SET_DEFAU LT	n	n	n	n	n	n
CONFIG_RA NDOM_TRUS T_BOOTLOA DER	y	y	N/D	N/D	N/D	N/D
CONFIG_RA NDOM_TRUS T_CPU	y	y	N/D	N/D	N/D	N/D
CONFIG_RE FCOUNT_FU LL	N/D	N/D	N/D	N/D	N/D	N/D
CONFIG_SC HED_CORE	N/D	y	N/D	y	N/D	y
CONFIG_SC HED_STACK _END_CHEC K	y	y	y	y	y	y
CONFIG_SE CCOMP	y	y	y	y	y	y
CONFIG_SE CCOMP_FIL TER	y	y	y	y	y	y

Opzione CONFIG	AL2023/6. 1/aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/aarch64	AL2023/6. 12/x86_64	AL2023/6. 18/aarch64	AL2023/6. 18/x86_64
<u>CONFIG_SECURITY</u>	y	y	y	y	y	y
<u>CONFIG_SECURITY_DMESG_RESTRICT</u>	y	y	y	y	y	y
<u>CONFIG_SECURITY_LANDLOCK</u>	y	y	y	y	y	y
<u>CONFIG_SECURITY_LOCKDOWN_LSM</u>	y	y	y	y	y	y
<u>CONFIG_SECURITY_LOCKDOWN_LSM_EARLY</u>	y	y	y	y	y	y
<u>CONFIG_SECURITY_SELINUX_BOOTPARAM</u>	y	y	y	y	y	y
<u>CONFIG_SECURITY_SELINUX_DEVLOP</u>	y	y	y	y	y	y

Opzione CONFIG	AL2023/6. 1/aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/aarch64	AL2023/6. 12/x86_64	AL2023/6. 18/aarch64	AL2023/6. 18/x86_64
CONFIG_SECURITY_SELINUX_DISABLE	n	n	N/D	N/D	N/D	N/D
CONFIG_SECURITY_WRITABLE_HOOKS	N/D	N/D	N/D	N/D	N/D	N/D
CONFIG_SECURITY_YAMA	y	y	y	y	y	y
CONFIG_SHUFFLE_PAGE_ALLOCATOR	y	y	y	y	y	y
CONFIG_SLAB_FREELIST_HARDENED	y	y	y	y	y	y
CONFIG_SLAB_FREELIST_RANDOM	y	y	y	y	y	y
CONFIG_SLUB_DEBUG	y	y	y	y	y	y
CONFIG_STACKPROTECTOR	y	y	y	y	y	y

Opzione CONFIG	AL2023/6. 1/aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/aarch64	AL2023/6. 12/x86_64	AL2023/6. 18/aarch64	AL2023/6. 18/x86_64
CONFIG_ST ACKPROTEC TOR_STRON G	y	y	y	y	y	y
CONFIG_ST ATIC_USER MODEHELPE R	n	n	n	n	n	n
CONFIG_ST RICT_DEVM EM	n	n	n	n	n	n
CONFIG_ST RICT_KERN EL_RWX	y	y	y	y	y	y
CONFIG_ST RICT_MODU LE_RWX	y	y	y	y	y	y
CONFIG_SY N_COOKIES	y	y	y	y	y	y
CONFIG_VM AP_STACK	y	y	y	y	y	y
CONFIG_WE RROR	n	n	n	n	n	n
CONFIG_ZE RO_CALL_U SED_REGS	n	n	n	n	n	n

Consenti ai metodi ACPI di essere in fase di esecuzione (CONFIG_ACPI_CUSTOM_METHOD) inserted/replaced

Amazon Linux disabilita questa opzione in quanto consente agli utenti `root` di scrivere nella memoria kernel arbitraria.

Questa opzione è una delle [impostazioni consigliate per Kernel Self Protection Project](#).

Formati binari vari (**binfmt_misc**)

Sebbene questa opzione sia una delle impostazioni consigliate dal [Kernel Self Protection Project \(KSPP\)](#), [AL2023 non imposta questa opzione di configurazione come consigliata da KSPP](#). In AL2023, questa funzionalità è opzionale ed è integrata come modulo del kernel.

Supporto di **BUG()**

Questa opzione è una delle [impostazioni consigliate per Kernel Self Protection Project](#).

BUG() se il kernel riscontra un danneggiamento dei dati durante la verifica della validità delle strutture di memoria del kernel

Alcune parti del kernel Linux controllano la coerenza interna delle strutture di dati e possono eseguire `BUG()` quando rilevano un danneggiamento dei dati.

Questa opzione è una delle [impostazioni consigliate per Kernel Self Protection Project](#).

COMPAT_BRK

Con questa opzione disabilitata (che è il modo in cui Amazon Linux configura il kernel), l'impostazione predefinita per `randomize_va_space` `sysctl` è 2, che abilita anche la randomizzazione degli heap sulla base `mmap`, sullo stack e sulla randomizzazione delle pagine VDSO.

Questa opzione esiste nel kernel per garantire la compatibilità con alcuni file binari `libc.so.5` obsoleti del 1996 e precedenti.

Questa opzione è una delle [impostazioni consigliate per Kernel Self Protection Project](#).

COMPAT_VDSO

Questa opzione di configurazione è rilevante per `x86-64` e non per `aarch64`. Impostandola su `n`, il kernel di Amazon Linux non rende visibile un Virtual Dynamic Shared Object (VDSO) a 32 bit a un

indirizzo prevedibile. La più recente libreria `glibc` nota per essere danneggiata dall'impostazione di questa opzione su `n` è `glibc 2.3.3`, del 2004.

Questa opzione è una delle [impostazioni consigliate per Kernel Self Protection Project](#).

CONFIG_DEBUGindurimento recintato

Le opzioni di configurazione del kernel Linux gated by `CONFIG_DEBUG` sono in genere progettate per l'uso in kernel creati per problemi di debug, e cose come le prestazioni non sono una priorità. AL2023 abilita l'opzione di rafforzamento. `CONFIG_DEBUG_LIST`

Disabilitazione di DMA per i dispositivi PCI nello stub EFI prima della configurazione di IOMMU

Sebbene questa opzione sia una delle [impostazioni consigliate dal Kernel Self Protection Project \(KSPP\)](#), AL2023 non imposta questa opzione di configurazione su quella consigliata da KSPP.

Rafforzamento per la copia della memoria tra kernel e spazio utente

Quando il kernel deve copiare la memoria da o verso lo spazio utente, questa opzione abilita alcuni controlli che possono proteggere da alcune classi di problemi di overflow dell'heap.

L'opzione `CONFIG_HARDENED_USERCOPY_FALLBACK` esisteva nei kernel da 4.16 a 5.15 per aiutare gli sviluppatori del kernel a scoprire eventuali voci mancanti dell'elenco degli indirizzi consentiti tramite un `WARN()`. Poiché AL2023 viene fornito un kernel 6.1, questa opzione non è più rilevante per. AL2023

L'`CONFIG_HARDENED_USERCOPY_PAGESPAN`opzione esisteva nei kernel principalmente come opzione di debug per gli sviluppatori e non si applica più al kernel 6.1 in. AL2023

Questa opzione è una delle [impostazioni consigliate per Kernel Self Protection Project](#).

Supporto per l'ibernazione

Sebbene questa opzione sia una delle impostazioni consigliate dal [Kernel Self Protection Project \(KSPP\)](#), AL2023 non imposta questa opzione di configurazione secondo quanto consigliato da KSPP. Questa opzione deve essere abilitata per supportare la possibilità di [ibernare l'istanza on demand](#) e quella di [ibernare le istanze spot interrotte](#)

Generazione di numeri casuali

Il kernel AL2023 è configurato per garantire che sia disponibile un'entropia adeguata per l'utilizzo all'interno di EC2.

CONFIG_INET_DIAG

Sebbene questa opzione sia una delle impostazioni consigliate dal [Kernel Self Protection Project \(KSPP\)](#), [AL2023 non imposta questa opzione di configurazione secondo quanto consigliato](#) da KSPP. In AL2023, questa funzionalità è opzionale ed è integrata come modulo del kernel.

Azzerata tutta la memoria del kernel, delle pagine e degli allocatori di slab durante l'allocazione e la deallocazione

Sebbene questa opzione sia una delle impostazioni consigliate dal [Kernel Self Protection Project \(KSPP\)](#), [AL2023 non imposta questa opzione di configurazione su quella consigliata](#) da KSPP. Queste opzioni sono disabilitate a AL2023 causa del possibile impatto sulle prestazioni derivante dall'attivazione di questa funzionalità per impostazione predefinita. Il comportamento `CONFIG_INIT_ON_ALLOC_DEFAULT_ON` può essere abilitato aggiungendolo `init_on_alloc=1` alla riga di comando del kernel e il comportamento `CONFIG_INIT_ON_FREE_DEFAULT_ON` può essere abilitato aggiungendo `init_on_free=1`.

Inizializzazione di tutte le variabili dello stack su zero
(**CONFIG_INIT_STACK_ALL_ZERO**)

Sebbene questa opzione sia una delle [impostazioni consigliate dal Kernel Self Protection Project \(KSPP\)](#), [AL2023 non imposta questa opzione di configurazione secondo quanto consigliato](#) da KSPP. Questa opzione ne richiede GCC 12 o più, mentre AL2023 viene fornita con GCC 11.

Firma del modulo Kernel

AL2023 firma e convalida le firme dei moduli del kernel. L'opzione `CONFIG_MODULE_SIG_FORCE`, che richiederebbe ai moduli di avere una firma valida, non è abilitata per preservare la compatibilità per gli utenti che creano moduli di terze parti. Per gli utenti che vogliono assicurarsi che tutti i moduli del kernel siano firmati, [Modulo di sicurezza Linux Lockdown \(LSM\)](#) può essere configurato in modo da imporre questa condizione.

kexec

Sebbene questa opzione sia una delle [impostazioni consigliate dal Kernel Self Protection Project \(KSPP\)](#), [AL2023 non imposta questa opzione di configurazione secondo quanto consigliato](#) da KSPP. Questa opzione è abilitata in modo da poter utilizzare la funzionalità kdump.

IOMMUSupport

AL2023 abilita il supporto IOMMU. L'opzione CONFIG_IOMMU_DEFAULT_DMA_STRICT non è abilitata per impostazione predefinita, ma questa funzionalità può essere configurata aggiungendo `iommu.passthrough=0 iommu.strict=1` alla riga di comando del kernel.

kfence

Sebbene questa opzione sia una delle [impostazioni consigliate dal Kernel Self Protection Project \(KSPP\)](#), [AL2023 non imposta questa opzione di configurazione su quella consigliata](#) da KSPP.

Supporto per **pty** legacy

AL2023 utilizza l'interfaccia moderna (). PTY devpts

Questa opzione è una delle [impostazioni consigliate per Kernel Self Protection Project](#).

Modulo di sicurezza Linux Lockdown (LSM)

AL2023 crea l'`lockdownLSM`, che bloccherà automaticamente il kernel quando si usa Secure Boot.

L'opzione CONFIG_LOCK_DOWN_KERNEL_FORCE_CONFIDENTIALITY non è abilitata. Sebbene questa opzione sia una delle [impostazioni consigliate dal Kernel Self Protection Project \(KSPP\)](#), [AL2023 non imposta questa opzione di configurazione secondo quanto consigliato](#) da KSPP. Quando non si utilizza Secure Boot, è possibile abilitare l'LSM Lockdown e configurarlo come desiderato.

Avvelenamento delle pagine

Sebbene questa opzione sia una delle [impostazioni consigliate dal Kernel Self Protection Project \(KSPP\)](#), [AL2023 non imposta questa opzione di configurazione su quella consigliata](#) da KSPP. Analogamente [Azzera tutta la memoria del kernel, delle pagine e degli allocatori di slab durante l'allocazione e la deallocazione](#), questa opzione è disabilitata nel AL2023 kernel a causa del possibile impatto sulle prestazioni.

Stack Protector

Il AL2023 kernel è costruito con la funzionalità stack-protector abilitata con l'GCCopzione. `-fstack-protector-strong`

Questa opzione è una delle [impostazioni consigliate per Kernel Self Protection Project](#).

seccomp BPF API

La funzionalità di rafforzamento seccomp viene utilizzata da software come systemd e i runtime dei container per rafforzare le applicazioni dello spazio utente.

Questa opzione è una delle [impostazioni consigliate per Kernel Self Protection Project](#).

Timeout **panic()**

Il AL2023 kernel è configurato con questo valore impostato su0, il che significa che il kernel non si riavvierà in caso di panico. Sebbene questa opzione sia una delle impostazioni consigliate dal [Kernel Self Protection Project \(KSPP\)](#), [AL2023 non imposta questa opzione di configurazione su quella consigliata](#) da KSPP. È configurabile tramite `sysctl`, `/proc/sys/kernel/panic` e sulla riga di comando del kernel.

Modelli di sicurezza

AL2023 abilita SELinux in modalità Permissiva per impostazione predefinita. Per ulteriori informazioni, consulta [Impostazione SELinux delle modalità per AL2 023](#).

Anche i moduli [Modulo di sicurezza Linux Lockdown \(LSM\)](#) e `yama` sono abilitati.

/proc/kcore

Sebbene questa opzione sia una delle [impostazioni consigliate dal Kernel Self Protection Project \(KSPP\)](#), [AL2023 non imposta questa opzione di configurazione su quella consigliata](#) da KSPP.

Randomizzazione dell'offset dello stack del kernel all'inserimento di `syscall`

Sebbene questa opzione sia una delle impostazioni consigliate dal [Kernel Self Protection Project \(KSPP\)](#), [AL2023 non imposta questa opzione di configurazione secondo quanto consigliato](#) da KSPP. Questa opzione può essere abilitata impostando `randomize_kstack_offset=on` sulla riga di comando del kernel.

Controlli di conteggio dei riferimenti (**CONFIG_REFCOUNT_FULL**)

Sebbene questa opzione sia una delle impostazioni consigliate dal [Kernel Self Protection Project \(KSPP\)](#), [AL2023 non imposta questa opzione di configurazione secondo quanto consigliato](#) da KSPP. Attualmente questa opzione non è abilitata a causa del suo possibile impatto sulle prestazioni.

Consapevolezza del pianificatore dei core SMT (**CONFIG_SCHED_CORE**)

Il AL2023 kernel è costruito con, il che consente l'utilizzo da parte delle applicazioni dello CONFIG_SCHED_CORE spazio utente. `prctl(PR_SCHED_CORE)` Questa opzione è una delle [impostazioni consigliate per Kernel Self Protection Project](#).

Verifica della presenza di danneggiamento dello stack durante le chiamate a **schedule()** (**CONFIG_SCHED_STACK_END_CHECK**)

Il AL2023 kernel è compilato con `enabled`. CONFIG_SCHED_STACK_END_CHECK Questa opzione è una delle [impostazioni consigliate per Kernel Self Protection Project](#).

Rafforzamento dell'allocatore di memoria

Il AL2023 kernel abilita il rafforzamento dell'allocatore di memoria del kernel con le opzioni,, and. CONFIG_SHUFFLE_PAGE_ALLOCATOR CONFIG_SLAB_FREELIST_HARDENED CONFIG_SLAB_FREELIST_RANDOM Questa opzione è una delle [impostazioni consigliate per Kernel Self Protection Project](#).

Supporto per il debug SLUB

Il AL2023 kernel abilita CONFIG_SLUB_DEBUG poiché questa opzione abilita funzionalità di debug opzionali per l'allocatore che possono essere abilitate sulla riga di comando del kernel. Questa opzione è una delle [impostazioni consigliate per Kernel Self Protection Project](#).

CONFIG_STATIC_USERMODEHELPER

Sebbene questa opzione sia una delle impostazioni consigliate dal [Kernel Self Protection Project \(KSPP\)](#), [AL2023 non imposta questa opzione di configurazione in base a quanto consigliato](#) da KSPP. Questo perché CONFIG_STATIC_USERMODEHELPER richiede un supporto speciale da parte della distribuzione, che attualmente non è presente in Amazon Linux.

Testo del kernel di sola lettura e rodata (**CONFIG_STRICT_KERNEL_RWX** e **CONFIG_STRICT_MODULE_RWX**)

Il AL2023 kernel è configurato per contrassegnare il testo e la memoria del kernel e dei moduli del kernel come di sola lettura e la rodata memoria non testuale contrassegnata come non eseguibile. Questa opzione è una delle [impostazioni consigliate per Kernel Self Protection Project](#).

Supporto per TCP syncookie (**CONFIG_SYN_COOKIES**)

Il AL2023 kernel è costruito con il supporto per i syncookie TCP. Questa opzione è una delle [impostazioni consigliate per Kernel Self Protection Project](#).

Stack mappato virtualmente con pagine guard (**CONFIG_VMAP_STACK**)

Il AL2023 kernel è costruito con `CONFIG_VMAP_STACK`, abilita stack di kernel mappati virtualmente con pagine guard. Questa opzione è una delle [impostazioni consigliate per Kernel Self Protection Project](#).

Compilazione tramite avvisi del compilatore come errori (**CONFIG_WERROR**)

Sebbene questa opzione sia una delle impostazioni consigliate dal [Kernel Self Protection Project \(KSPP\)](#), [AL2023 non imposta questa opzione di configurazione come consigliata](#) da KSPP.

Registrazione dell'azzeramento sulla funzione exit (**CONFIG_ZERO_CALL_USED_REGS**)

Sebbene questa opzione sia una delle impostazioni consigliate dal [Kernel Self Protection Project \(KSPP\)](#), [AL2023 non imposta questa opzione di configurazione secondo quanto consigliato](#) da KSPP.

Indirizzo minimo per l'allocazione dello spazio utente

Questa opzione di rafforzamento può aiutare a ridurre l'impatto dei bug dei puntatori NULL del kernel. Questa opzione è una delle [impostazioni consigliate per Kernel Self Protection Project](#).

clang opzioni di indurimento specifiche

Il AL2023 kernel è compilato con GCC invece di clang, quindi l'opzione di `CONFIG_CFI_CLANG` rafforzamento non può essere abilitata, il che rende `CONFIG_CFI_PERMISSIVE` inoltre non

applicabile. Sebbene questa opzione sia una delle [impostazioni consigliate dal Kernel Self Protection Project \(KSPP\)](#), AL2023 non imposta questa opzione di configurazione su quella consigliata da KSPP.

Opzioni di rafforzamento del kernel specifiche di x86-64

Opzione CONFIG	AL2023/6. 1/aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/aarch64	AL2023/6. 12/x86_64	AL2023/6. 18/aarch64	AL2023/6. 18/x86_64
CONFIG_AMD_IOMMU	N/D	y	N/D	y	N/D	y
CONFIG_AMD_IOMMU_V2	N/D	y	N/D	N/D	N/D	N/D
CONFIG_IA32_EMULATION	N/D	y	N/D	y	N/D	y
CONFIG_INTEL_IOMMU	N/D	y	N/D	y	N/D	y
CONFIG_INTEL_IOMMU_DEFAULT_ON	N/D	n	N/D	n	N/D	n
CONFIG_INTEL_IOMMU_SVM	N/D	n	N/D	n	N/D	n
CONFIG_LEGACY_VSYSCALL_NONE	N/D	n	N/D	n	N/D	n

Opzione CONFIG	AL2023/6. 1/aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/aarch64	AL2023/6. 12/x86_64	AL2023/6. 18/aarch64	AL2023/6. 18/x86_64
CONFIG_MODIFY_LDT_SYSCALL	N/D	n	N/D	n	N/D	n
CONFIG_PAGE_TABLE_ISOLATION	N/D	y	N/D	N/D	N/D	N/D
CONFIG_RANDOMIZE_MEMORY	N/D	y	N/D	y	N/D	y
CONFIG_X86_64	N/D	y	N/D	y	N/D	y
CONFIG_X86_64_MSR	N/D	y	N/D	y	N/D	y
CONFIG_X86_64_VSYSCALL_EMULATION	N/D	y	N/D	y	N/D	y
CONFIG_X86_64_X32	N/D	N/D	N/D	N/D	N/D	N/D
CONFIG_X86_64_X32_ABI	N/D	n	N/D	n	N/D	n

Supporto per x86-64

Il supporto base per x86-64 include il supporto per i bit Physical Address Extension (PAE) e No eXecute (NX). Questa opzione è una delle [impostazioni consigliate per Kernel Self Protection Project](#).

Supporto per AMD e Intel IOMMU

Il AL2023 kernel viene compilato con il supporto per AMD e Intel IOMMUs. Questa opzione è una delle [impostazioni consigliate per Kernel Self Protection Project](#).

L'opzione `CONFIG_INTEL_IOMMU_DEFAULT_ON` non è impostata, ma può essere abilitata passando `intel_iommu=on` alla riga di comando del kernel. Sebbene questa opzione sia una delle [impostazioni consigliate dal Kernel Self Protection Project \(KSPP\)](#), AL2023 non imposta questa opzione di configurazione secondo quanto consigliato da KSPP.

L'opzione `CONFIG_INTEL_IOMMU_SVM` non è attualmente abilitata in AL2023. Sebbene questa opzione sia una delle [impostazioni consigliate dal Kernel Self Protection Project \(KSPP\)](#), AL2023 non imposta questa opzione di configurazione su quella consigliata da KSPP.

Supporto per lo spazio utente a 32 bit

Important

Il supporto per lo spazio utente x86 a 32 bit è obsoleto e il supporto per l'esecuzione di file binari dello spazio utente a 32 bit potrebbe essere rimosso in una futura versione principale di Amazon Linux.

Note

Sebbene AL2023 non includa più pacchetti a 32 bit, il kernel supporterà comunque l'esecuzione di uno spazio utente a 32 bit. Per ulteriori informazioni, consulta [Pacchetti x86 \(i686\) a 32 bit](#).

Per supportare l'esecuzione di applicazioni userspace a 32 bit, AL2023 non abilita l'opzione `CONFIG_X86_VSYSCALL_EMULATION` e abilita le opzioni, and. `CONFIG_IA32_EMULATION` `CONFIG_COMPAT` `CONFIG_X86_VSYSCALL_EMULATION`. Sebbene questa opzione sia una delle [impostazioni consigliate dal Kernel Self Protection Project \(KSPP\)](#), AL2023 non imposta questa opzione di configurazione su quella consigliata da KSPP.

L'ABI x32 nativo a 32 bit per processori a 64 bit non è abilitato (`CONFIG_X86_X32` e `CONFIG_X86_X32_ABI`). Questa opzione è una delle [impostazioni consigliate per Kernel Self Protection Project](#).

Supporto per Model Specific Register x86 (MSR)

L'opzione `CONFIG_X86_MSR` è abilitata per supportare turbostat. Sebbene questa opzione sia una delle impostazioni consigliate dal [Kernel Self Protection Project \(KSPP\)](#), AL2023 non imposta questa opzione di configurazione secondo quanto consigliato da KSPP.

syscall `modify_ldt`

AL2023 non consente ai programmi utente di modificare la Local Descriptor Table (LDT) x86 con `syscall.modify_ldt`. Questa chiamata è necessaria per eseguire codice a 16 bit o segmentato e la sua assenza può compromettere software come `dosemu`, l'esecuzione di alcuni programmi in `WINE` e alcune librerie di threading molto vecchie. Questa opzione è una delle [impostazioni consigliate per Kernel Self Protection Project](#).

Rimozione della mappatura del kernel in modalità utente

AL2023 configura il kernel in modo che la maggior parte degli indirizzi del kernel non sia mappata nello spazio utente. Questa opzione è una delle [impostazioni consigliate per Kernel Self Protection Project](#).

Randomizzazione delle sezioni di memoria del kernel

AL2023 configura il kernel per randomizzare gli indirizzi virtuali di base delle sezioni di memoria del kernel. Questa opzione è una delle [impostazioni consigliate per Kernel Self Protection Project](#).

Opzioni di rafforzamento del kernel specifiche di aarch64

Opzione CONFIG	AL2023/6. 1/aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/aarch64	AL2023/6. 12/x86_64	AL2023/6. 18/aarch64	AL2023/6. 18/x86_64
CONFIG_ARM64_BTI	y	N/D	y	N/D	y	N/D
CONFIG_ARM64_BTI_KERNEL	N/D	N/D	N/D	N/D	N/D	N/D
CONFIG_ARM64_PTR_AUTH	y	N/D	y	N/D	y	N/D

Opzione CONFIG	AL2023/6. 1/aarch64	AL2023/6. 1/x86_64	AL2023/6. 12/aarch64	AL2023/6. 12/x86_64	AL2023/6. 18/aarch64	AL2023/6. 18/x86_64
CONFIG_ARM64_PTR_AUTH_KERNEL	y	N/D	y	N/D	y	N/D
CONFIG_ARM64_SW_TTBR0_PAN	y	N/D	y	N/D	y	N/D
CONFIG_UNMAP_KERNEL_L_AT_EL0	y	N/D	y	N/D	y	N/D

Identificazione dei target di ramo

Il AL2023 kernel abilita il supporto per Branch Target Identification (). `CONFIG_ARM64_BTI` Questa opzione è una delle [impostazioni consigliate per Kernel Self Protection Project](#).

[L'CONFIG_ARM64_BTI_KERNEL](#) opzione non è abilitata in AL2023 quanto è stata compilata con GCC, e il supporto per la compilazione del kernel con questa opzione è attualmente disabilitato nel kernel upstream a causa di un bug di gcc. Sebbene questa opzione sia una delle impostazioni consigliate dal [Kernel Self Protection Project \(KSPP\)](#), AL2023 non imposta questa opzione di configurazione in base a quanto consigliato da KSPP.

Autenticazione dei puntatori (**CONFIG_ARM64_PTR_AUTH**)

Il AL2023 kernel è costruito con il supporto per l'estensione Pointer Authentication (parte delle estensioni ARMv8 .3), che può essere utilizzata per aiutare a mitigare le tecniche ROP (Return Oriented Programming). Il supporto hardware richiesto per l'autenticazione dei puntatori su [Graviton](#) è stato introdotto con Graviton 3.

L'opzione `CONFIG_ARM64_PTR_AUTH` è abilitata e fornisce supporto per l'autenticazione dei puntatori per lo spazio utente. Poiché anche l'`CONFIG_ARM64_PTR_AUTH_KERNEL` opzione è abilitata, il AL2023 kernel è in grado di utilizzare autonomamente la protezione dell'indirizzo di ritorno.

Questa opzione è una delle [impostazioni consigliate per Kernel Self Protection Project](#).

Emulazione dell'accesso con privilegi che non usa mai lo switching `TTBR0_EL1`

Questa opzione impedisce al kernel di accedere direttamente alla memoria dello spazio utente, con `TTBR0_EL1` che viene impostato solo temporaneamente su un valore valido dalle routine di accesso utente.

Questa opzione è una delle [impostazioni consigliate per Kernel Self Protection Project](#).

Annullamento della mappatura del kernel durante l'esecuzione nello spazio utente

Il AL2023 kernel è configurato per demappare il kernel quando viene eseguito in userspace (`CONFIG_UNMAP_KERNEL_AT_EL0`). Questa opzione è una delle [impostazioni consigliate per Kernel Self Protection Project](#).

Accesso ai metadati del repository AL2023

A partire dalla release `2023.11.20260406`, i AL2023 repository includono firme crittografiche per i metadati dei repository. Ogni file del repository è accompagnato da un `repomd.xml` file di firma GPG separato (`repomd.xml.asc`) che è possibile utilizzare per verificare l'autenticità e l'integrità dei metadati del repository prima che i pacchetti vengano scaricati.

Questa firma si aggiunge alla firma del pacchetto RPM esistente (`gpgcheck`), che verifica i singoli pacchetti. La firma dei metadati del repository verifica i metadati che descrivono il contenuto del repository, come l'elenco dei pacchetti disponibili e i relativi checksum.

Come funziona la firma dei metadati nel repository

Quando i AL2023 repository vengono pubblicati, i metadati del repository (`repomd.xml`) vengono firmati utilizzando una chiave KMS. AWS La firma distaccata risultante (`repomd.xml.asc`) viene posizionata accanto ai metadati nel repository.

Quando la abiliti `repo_gpgcheck` nella configurazione del repository, scarica e verifica DNF automaticamente la `repomd.xml.asc` firma rispetto alla chiave pubblica GPG prima di utilizzare i metadati del repository. Se la verifica della firma fallisce, DNF rifiuta i metadati del repository e non procede con le operazioni sui pacchetti da quel repository. [Per ulteriori informazioni in merito a `repo_gpgcheck`, consulta il Configuration Reference.](#) DNF

I seguenti AL2023 repository includono metadati firmati:

- Archivio principale (`amazonlinux`)

- Archivio Kernel Livepatch () `kernel-livepatch`
- Archivio NVIDIA () `amazonlinux-nvidia`
- Pacchetti supplementari per il repository Amazon Linux () `amazonlinux-spa1`

Differenza tra `gpgcheck` e `repo_gpgcheck`

Impostazione	Cosa verifica	Impostazione predefinita in AL2023
<code>gpgcheck=1</code>	Verifica la firma GPG dei singoli pacchetti RPM prima dell'installazione.	Abilitato
<code>repo_gpgcheck=1</code>	Verifica la firma GPG dei metadati del repository () prima di utilizzare il repository. <code>repomd.xml</code>	Disabilitato (abilitato di default a partire dalla versione trimestrale) <code>2023.12</code>

Ti consigliamo vivamente di abilitare entrambi `gpgcheck` e `repo_gpgcheck`. Ciò garantisce che sia i metadati del repository che i singoli pacchetti vengano verificati prima dell'uso.

Abilitazione della verifica dei metadati del repository

È possibile abilitare la verifica dei metadati del repository per i singoli repository aggiornando i relativi file di configurazione.

Important

A partire dalla versione `2023.12` trimestrale, `repo_gpgcheck=1` sarà abilitata per impostazione predefinita nei file di configurazione del repository. AL2023

Abilita per un repository specifico

I file di configurazione AL2023 del repository sono `/etc/yum.repos.d/` impostati di `repo_gpgcheck=0` default. Per abilitare la verifica dei metadati del repository, modifica questo valore `1` nella configurazione del repository. Ad esempio, per abilitarlo per il repository principale:

```
[amazonlinux]
name=Amazon Linux 2023 repository
...
gpgcheck=1
repo_gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-amazon-linux-2023
```

Verifica del funzionamento della firma dei metadati del repository

Dopo l'attivazione `repo_gpgcheck=1`, puoi verificare che la verifica dei metadati funzioni svuotando la DNF cache e aggiornando i metadati:

```
[ec2-user ~]$ sudo dnf clean metadata
[ec2-user ~]$ sudo dnf makecache
```

Se la verifica dei metadati ha esito positivo, DNF importa la chiave GPG (se non è già stata importata) e crea la cache dei metadati senza errori. Verrà visualizzato un output simile al seguente:

```
Amazon Linux 2023 repository          1.7 MB/s | 1.8 kB    00:00
Importing GPG key 0xD832C631:
  Userid      : "Amazon Linux <amazon-linux@amazon.com>"
  Fingerprint: B21C 50FA 44A9 9720 EAA7 2F7F E951 904A D832 C631
  From        : /etc/pki/rpm-gpg/RPM-GPG-KEY-amazon-linux-2023
Amazon Linux 2023 repository          18 MB/s | 55 MB     00:03
Metadata cache created.
```

Se la verifica della firma fallisce, DNF visualizza un messaggio di errore che indica un errore di verifica della firma GPG e la creazione della cache dei metadati non riesce.

Chiavi pubbliche GPG per i repository AL2023

Le chiavi pubbliche GPG utilizzate per la verifica dei metadati del repository vengono installate dalla configurazione del repository corrispondente su. RPMs `/etc/pki/rpm-gpg/` La tabella seguente elenca le chiavi pubbliche utilizzate da ciascun repository.

Repository	Chiave di firma del pacchetto	Chiave di firma Repodata	Distribuito in
Nucleo (amazonlinux)	RPM-GPG-KEY-amazon-linux-2023	RPM-GPG-KEY-amazon-linux-2023	system-release
Kernel Livepatch () kernel-livepatch	RPM-GPG-KEY-amazon-linux-2023	RPM-GPG-KEY-amazon-linux-2023	system-release
NVIDIA () amazonlinux-nvidia	RPM-GPG-KEY-NVIDIA-D42D0685	RPM-GPG-KEY-amazon-linux-2023-nvidia	nvidia-release
SPAL () amazonlinux-spal	RPM-GPG-KEY-amazon-linux-spal	RPM-GPG-KEY-amazon-linux-spal	spal-release

Queste chiavi vengono installate automaticamente quando si installa l'RPM di configurazione del repository corrispondente.

Avvio sicuro UEFI attivo AL2023

AL2023 supporta UEFI Secure Boot a partire dalla versione 2023.1. Devi utilizzare AL2023 con istanze Amazon EC2 che supportano sia UEFI che UEFI Secure Boot. Per ulteriori informazioni, consulta [Requisiti per il lancio di un'istanza Amazon EC2 in modalità di avvio UEFI](#) nella Amazon EC2 User Guide.

AL2023 le istanze con UEFI Secure Boot abilitato accettano solo codice a livello di kernel, incluso il kernel Linux e i moduli, che sono firmati da, Amazon così puoi assicurarti che l'istanza esegua solo codici a livello di kernel firmati da. AWS

Per ulteriori informazioni sulle istanze Amazon EC2 e UEFI Secure Boot, consulta [UEFI Secure Boot for Amazon EC2 nella Amazon EC2 User Guide](#).

Prerequisiti

- È necessario utilizzare un'AMI con AL2023 versione 2023.1 o successiva.
- Il tipo di istanza deve supportare UEFI Secure Boot. Per ulteriori informazioni, consulta [Requisiti per il lancio di un'istanza Amazon EC2 in modalità di avvio UEFI](#) nella Amazon EC2 User Guide.

Abilita UEFI Secure Boot su AL2023

Lo standard AL2023 AMIs incorpora un bootloader e un kernel firmati dalle nostre chiavi. È possibile abilitare UEFI Secure Boot registrando le istanze esistenti o creando AMIs con UEFI Secure Boot preabilitato registrando un'immagine da un'istantanea. UEFI Secure Boot non è abilitato per impostazione predefinita sullo standard. AL2023 AMIs

La modalità di avvio di AL2023 AMIs è impostata in modo da garantire `uefi-preferred` che le istanze avviate con questi dispositivi AMIs utilizzino il firmware UEFI, se il tipo di istanza supporta UEFI. Se il tipo di istanza non supporta UEFI, l'istanza viene avviata con il firmware BIOS legacy. Quando un'istanza viene avviata in modalità BIOS legacy, UEFI Secure Boot non viene applicato.

Per ulteriori informazioni sulle modalità di avvio AMI sulle istanze Amazon EC2, consulta il [comportamento di avvio delle istanze con le modalità di avvio di Amazon EC2 nella Amazon EC2 User Guide](#).

Argomenti

- [Registrazione di un'istanza esistente](#)
- [Registrazione di un'immagine dallo snapshot](#)
- [Aggiornamenti di revoca](#)
- [Come funziona UEFI Secure Boot su AL2023](#)
- [Registrazione di chiavi personalizzate](#)

Registrazione di un'istanza esistente

Per registrare un'istanza esistente, compila le variabili specifiche del firmware UEFI con un set di chiavi che consentono al firmware di verificare il bootloader e al bootloader di verificare il kernel all'avvio successivo.

1. Amazon Linux fornisce uno strumento per semplificare il processo di registrazione. Esegui il comando seguente per effettuare il provisioning dell'istanza con il set di chiavi e certificati necessario.

```
sudo amazon-linux-sb enroll
```

2. Esegui il seguente comando per riavviare l'istanza. Dopo il riavvio dell'istanza, verrà abilitato UEFI Secure Boot.

```
sudo reboot
```

Note

Amazon Linux AMIs attualmente non supporta Nitro Trusted Platform Module (NitroTPM). Se hai bisogno di NitroTPM oltre a UEFI Secure Boot, usa le informazioni nella sezione seguente.

Registrazione di un'immagine dallo snapshot

Quando registri un'AMI da uno snapshot di un volume root di Amazon EBS utilizzando l'API `register-image` Amazon EC2, puoi effettuare il provisioning dell'AMI con un blob binario che contiene lo stato dell'archivio di variabili UEFI. Fornendo `AL2023 UefiData`, abiliti UEFI Secure Boot e non devi seguire i passaggi della sezione precedente.

Per ulteriori informazioni sulla creazione e l'utilizzo di un blob binario, consulta [Creare un blob binario contenente un archivio di variabili precompilate](#) nella Amazon EC2 User Guide.

AL2023 fornisce un blob binario predefinito che può essere utilizzato direttamente sulle istanze Amazon EC2. Il blob binario si trova in `/usr/share/amazon-linux-sb-keys/uefi.vars` su un'istanza in esecuzione. Questo blob è fornito dal pacchetto `amazon-linux-sb-keys` RPM che viene installato di default a partire dalla versione 2023.1. AL2023 AMIs

Note

Per assicurarti di utilizzare la versione più recente delle chiavi e delle revoche, usa il blob della stessa versione AL2023 che usi per creare l'AMI.

Quando registri un'immagine, si consiglia di utilizzare il parametro `BootMode` dell'API [RegisterImage](#) impostata su `uefi`. Ciò consente di abilitare NitroTPM impostando il parametro `TpmSupport` su `v2.0`. Inoltre, impostando `BootMode` su `uefi` è possibile garantire che UEFI Secure Boot sia abilitato e non possa essere disabilitato accidentalmente quando si passa a un tipo di istanza che non supporta UEFI.

Per ulteriori informazioni su NitroTPM, consulta [NitroTPM per istanze Amazon EC2 nella Amazon EC2 User Guide](#).

Aggiornamenti di revoca

Potrebbe essere necessario che Amazon Linux distribuisca una nuova versione del bootloader `grub2` o del kernel Linux firmata con chiavi aggiornate. In tal caso, potrebbe essere necessario revocare la vecchia chiave per evitare la possibilità che bug sfruttabili delle versioni precedenti del bootloader possano aggirare il processo di verifica di UEFI Secure Boot.

Gli aggiornamenti dei pacchetti ai pacchetti `grub2` o `kernel` aggiornano sempre automaticamente l'elenco delle revoche nell'archivio di variabili UEFI dell'istanza in esecuzione. Ciò significa che con UEFI Secure Boot abilitato, non è più possibile eseguire la versione obsoleta di un pacchetto dopo aver installato un aggiornamento di sicurezza per il pacchetto.

Come funziona UEFI Secure Boot su AL2023

A differenza di altre distribuzioni Linux, Amazon Linux non fornisce un componente aggiuntivo, chiamato shim, che funge da bootloader di prima fase. Lo shim è generalmente firmato con chiavi Microsoft. Ad esempio, nelle distribuzioni Linux con shim, lo shim carica il bootloader `grub2` che usa il codice dello shim per verificare il kernel Linux. Inoltre, lo shim mantiene il proprio set di chiavi e revoche nel database Machine Owner Key (MOK) situato nell'archivio di variabili UEFI e controllato con lo strumento `mokutil`.

Amazon Linux non fornisce uno shim. Poiché il proprietario dell'AMI controlla le variabili UEFI, questo passaggio intermedio non è necessario e può influire negativamente sui tempi di avvio e lancio. Inoltre, per impostazione predefinita, abbiamo scelto di non includere l'attendibilità nelle chiavi di tutti i fornitori, per ridurre la possibilità che i file binari indesiderati possano essere eseguiti. Come sempre, i clienti possono includere file binari se lo desiderano.

Con Amazon Linux, UEFI carica e verifica direttamente il nostro bootloader `grub2`. Il bootloader `grub2` è stato modificato per utilizzare UEFI per verificare il kernel Linux dopo averlo caricato. Pertanto, il kernel Linux viene verificato utilizzando gli stessi certificati memorizzati nella normale

variabile db UEFI (database delle chiavi autorizzate) e testato rispetto alla stessa variabile dbx (database delle revoche) del bootloader e di altri file binari UEFI. Poiché forniamo le nostre chiavi PK e KEK, che controllano l'accesso al database db e al database dbx, possiamo distribuire le revoche e gli aggiornamenti firmati secondo necessità senza un intermediario come lo shim.

Per ulteriori informazioni su UEFI Secure Boot, consulta [Come funziona UEFI Secure Boot con le istanze Amazon EC2 nella Amazon EC2 User Guide](#).

Registrazione di chiavi personalizzate

Come documentato nella sezione precedente, Amazon Linux non richiede uno shim per UEFI Secure Boot su Amazon EC2. Quando leggi la documentazione per altre distribuzioni Linux, potresti trovare la documentazione per la gestione del database Machine Owner Key (MOK), che non è presente su `mokutil`. AL2023 Gli ambienti shim e MOK aggirano alcune limitazioni della registrazione delle chiavi nel firmware UEFI che non sono applicabili al modo in cui Amazon EC2 implementa UEFI Secure Boot. Con Amazon EC2 sono disponibili meccanismi per manipolare facilmente e direttamente le chiavi nell'archivio di variabili UEFI.

Se desideri registrare le tue chiavi, puoi farlo manipolando l'archivio delle variabili all'interno di un'istanza esistente (vedi [Aggiungere chiavi all'archivio delle variabili dall'interno dell'istanza](#)) o costruendo un blob binario precompilato (vedi [Creare un blob binario contenente un archivio di variabili precompilato](#)).

Le traduzioni sono generate tramite traduzione automatica. In caso di conflitto tra il contenuto di una traduzione e la versione originale in Inglese, quest'ultima prevarrà.