



Diagram Arsitektur

Hosting Web Tanpa Server di AWS App Runner



Hosting Web Tanpa Server di AWS App Runner: Diagram Arsitektur

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

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

Table of Contents

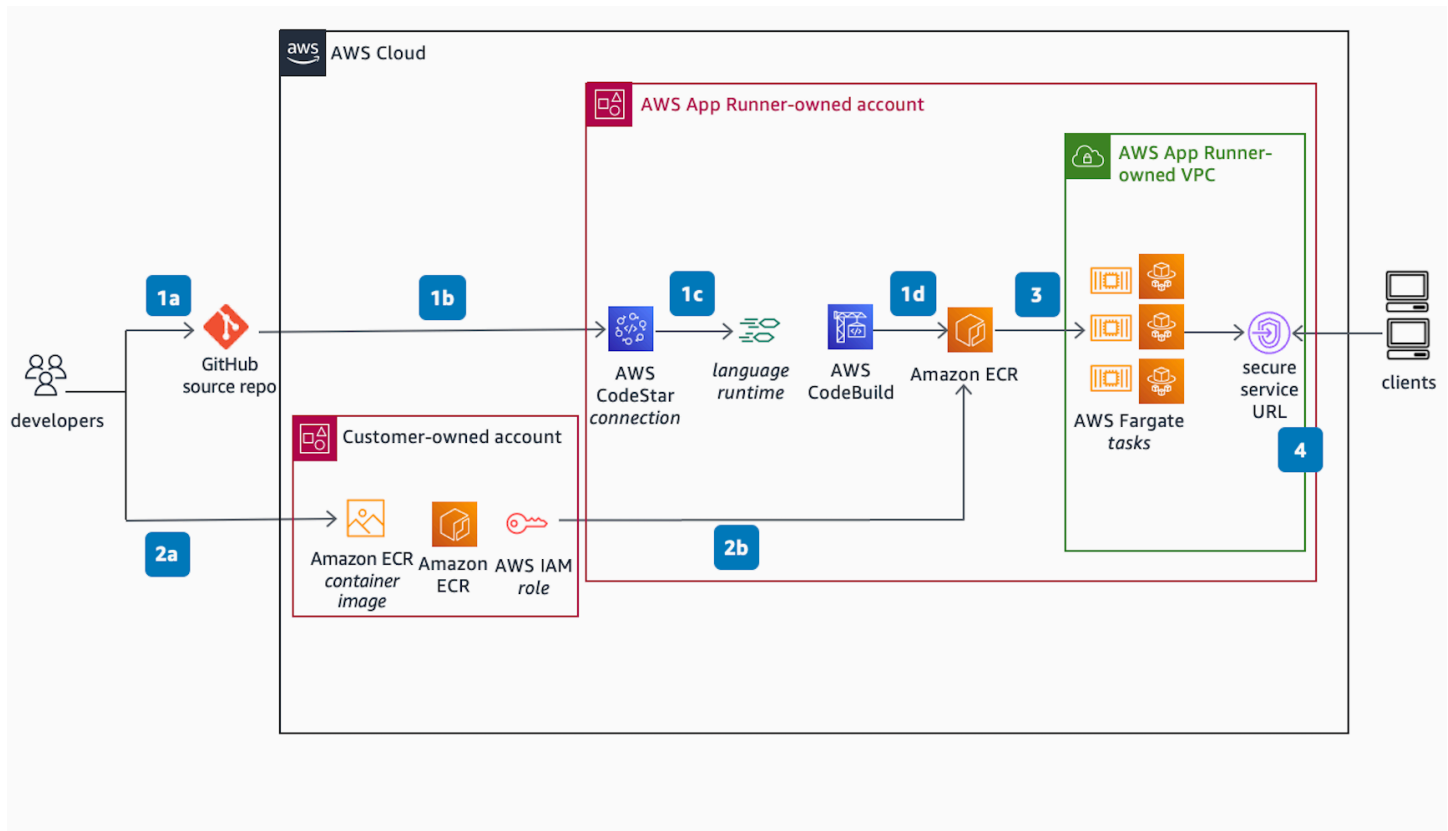
Hosting Web Tanpa Server aktif AWS App Runner	i
Hosting Web Tanpa Server aktif AWS App Runner	1
Hosting Web Tanpa Server aktif AWS App Runner	3
Sumber bacaan lebih lanjut	4
Sejarah diagram	4
.....	vi

Hosting Web Tanpa Server aktif AWS App Runner

Tanggal publikasi: 3 Mei 2023 ([Sejarah diagram](#))

Arsitektur referensi ini merinci dua skenario tentang cara menjalankan aplikasi web kontainer tanpa server tanpa perlu menyediakan atau mengelola infrastruktur.

Hosting Web Tanpa Server aktif AWS App Runner



Pengalaman pengembang: Gunakan AWS App Runner untuk membuat dan mengelola layanan web berdasarkan dua jenis sumber layanan: kode sumber dan gambar sumber.

1. Skenario 1 menggunakan kode sumber:

- Komit kode ke repo GitHub sumber.
- Otorisasi akses ke kode sumber dalam GitHub penggunaan AWS CodeStar.

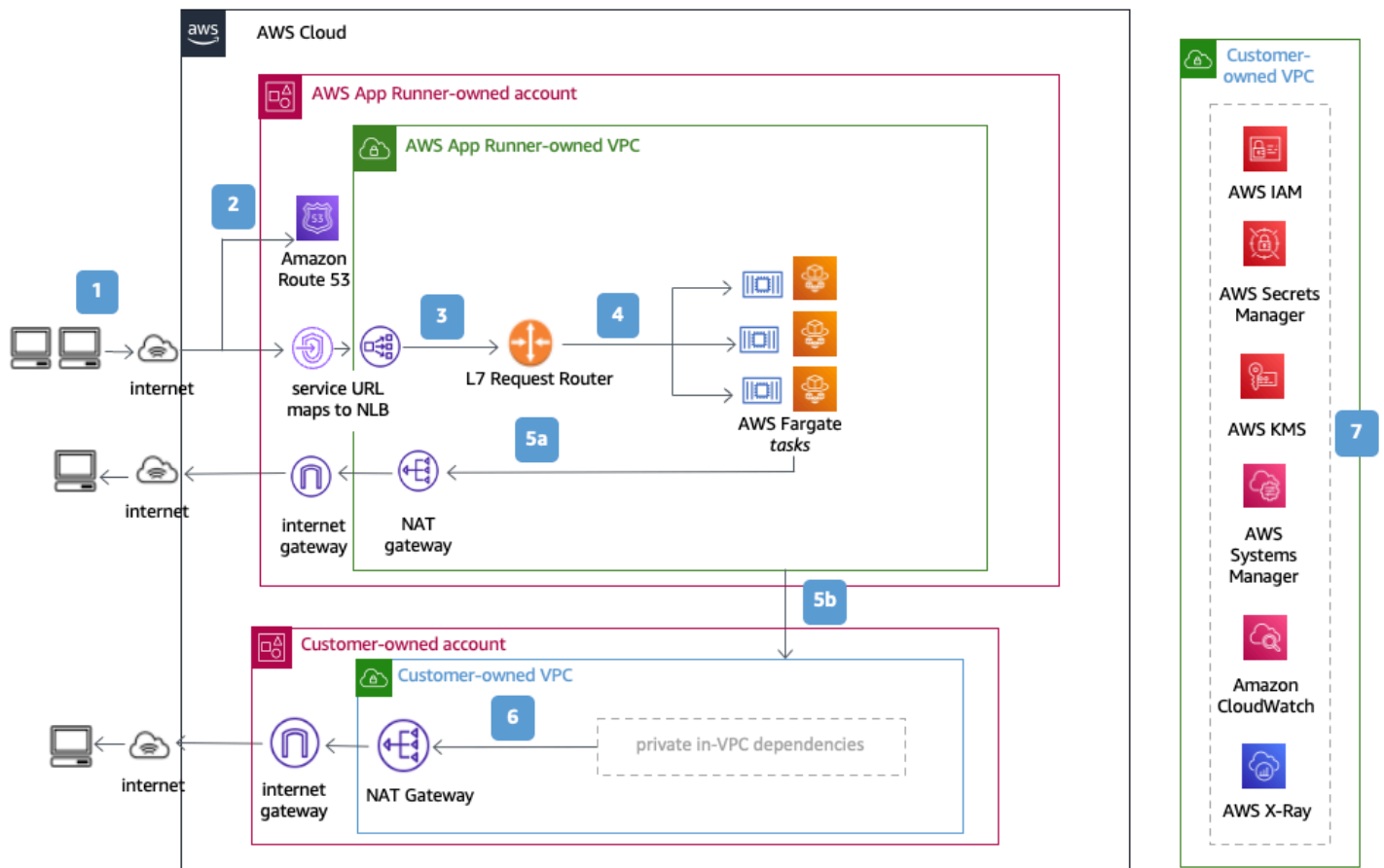
- c. Berikan instruksi untuk membangun dan menjalankan layanan web dan menentukan lingkungan runtime terkelola yang didukung oleh AWS App Runner. AWS CodeBuild kemudian mengemas kode dan dependensinya secara otomatis dan membangun image kontainer baru.
 - d. Gambar wadah kemudian didorong dan disimpan ke dalam repositori Amazon Elastic Container Registry (Amazon ECR), yang dihosting di dalam akun yang AWS App Runner dimiliki.
2. Skenario 2 menggunakan gambar sumber:
- a. Atau, mulailah dengan gambar pra-buat yang disimpan di repositori Amazon ECR yang dikelola pelanggan, yang dihosting di dalam akun milik pelanggan.

b. Dengan mengotorisasi akses ke gambar kontainer menggunakan AWS IAM Identity Center Role, gambar kemudian didorong dan disimpan ke repositori Amazon ECR yang AWS App Runner dikelola.

Pembuatan Layanan App Runner:

3. Gambar kontainer yang disimpan di Amazon ECR kemudian ditarik untuk digunakan sebagai *Tu* AWS Fargate *gas* tanpa server di dalam Amazon Virtual Private Cloud yang AWS App Runner dimiliki, juga dikenal sebagai AWS App Runner Service VPC. *Tu* AWS Fargate *gas* diterapkan pada AWS Fargate cluster yang berjalan di beberapa Zona Ketersediaan untuk ketersediaan tinggi di subnet pribadi, dan dikonfigurasi dengan Penskalaan Otomatis AWS *atis* Aplikasi.
4. URL layanan aman adalah *created/published* setelah penerapan berhasil. URL Layanan Aman memetakan ke AWS Network Load Balancer yang menghadap publik dan diberi nama domain default oleh AWS App Runner Pelanggan juga dapat memiliki nama domain khusus *associate/map* yang mereka miliki, tetapi harus dapat diselesaikan secara publik dan dapat didaftarkan dengan Amazon Route 53 Public Hosted Zone atau dengan penyedia DNS mana pun.

Hosting Web Tanpa Server aktif AWS App Runner



AWS App Runner adalah solusi yang aman dan konsisten untuk mengekspos aplikasi web menggunakan titik akhir publik atau URL layanan.

Jalur lalu lintas masuk:

1. Klien memulai permintaan ke URL layanan.
2. URL layanan diselesaikan ke Network Load Balancer yang menghadap publik menggunakan Amazon Route 53. AWS NLB secara otomatis disediakan dan dimiliki oleh layanan. AWS App Runner
3. Permintaan memasuki VPC AWS App Runner milik melalui AWS Network Load Balancer (AWS NLB) dan dialihkan ke Router Permintaan L7.
4. Router Permintaan L7 adalah merutekan permintaan masuk ke instance layanan web tertentu atau AWS Fargate Tugas.

Jalur lalu lintas keluar, Skenario 1 (layanan web tidak memerlukan akses ke dependensi hilir dalam customer/private VPC):

5. a. Lalu lintas keluar (atau kembali) dialihkan ke internet melalui NAT Gateway yang AWS dikelola dan gateway internet yang disediakan dalam VPC AWS App Runner milik, atau mode jaringan default.

Jalur Lalu Lintas Keluar, Skenario 2 (layanan web memerlukan akses ke dependensi hilir dalam customer/private VPC):

- b. Lalu lintas keluar diteruskan ke VPC customer/private milik -dimana private resources/dependencies disediakan secara manual oleh end-. user/developer Lalu lintas diteruskan ke titik akhir tujuan per tabel routing VPC.
6. Lalu lintas ke internet dialihkan melalui NAT Gateway yang dikelola pelanggan dan gateway internet yang disediakan dalam VPC customer/private milik, atau mode jaringan VPC Egress.
7. AWS Identity and Access Management(AWS IAM), AWS Key Management Service (AWS KMS) AWS Secrets Manager, dan AWS Systems Manager Parameter Store memastikan akses berbasis peran dan menyimpan data rahasia dengan aman. Amazon CloudWatch dan per AWS X-Ray tahankan observabilitas.

Sumber bacaan lebih lanjut

Untuk informasi tambahan, lihat

- [AWS Ikon Arsitektur](#)
- [AWS Pusat Arsitektur](#)
- [AWS Well-Architected](#)

Sejarah diagram

Untuk diberitahu tentang pembaruan diagram arsitektur referensi ini, berlangganan umpan RSS.

Perubahan	Deskripsi	Tanggal
Publikasi awal	Diagram arsitektur referensi pertama kali diterbitkan.	3 Mei 2021

 Note

Untuk berlangganan pembaruan RSS, Anda harus mengaktifkan plugin RSS untuk browser yang Anda gunakan.

Terjemahan disediakan oleh mesin penerjemah. Jika konten terjemahan yang diberikan bertentangan dengan versi bahasa Inggris aslinya, utamakan versi bahasa Inggris.