



Panduan Developer FleetIQ

# Amazon GameLift



Versi

---

# Amazon GameLift: Panduan Developer FleetIQ

Copyright © 2024 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 menghina atau mendiskreditkan Amazon. Semua merek dagang lain yang tidak dimiliki oleh Amazon merupakan properti dari masing-masing pemilik, yang mungkin berafiliasi, terkait dengan, atau disponsori oleh Amazon, atau tidak.

---

# Table of Contents

Apa itu Amazon GameLift FleetIQ? .....	1
Cara kerja FleetIQ .....	2
Logika Amazon GameLift FleetIQ .....	2
Sumber daya dan komponen utama .....	5
Arsitektur game .....	7
Melengkapi hosting on-premises .....	7
Siklus hidup dan status .....	9
Proses penyeimbangan spot .....	9
Kehidupan grup server game .....	12
Kehidupan server game .....	14
Praktik terbaik .....	17
Fitur Amazon GameLift FleetIQ .....	20
Harga untuk Amazon GameLift FleetIQ .....	21
Mengatur .....	22
Perangkat lunak yang didukung .....	22
Siapkan AWS akun Anda .....	23
Buat sebuah Akun AWS .....	23
Mengelola izin pengguna untuk Amazon GameLift FleetIQ .....	25
Membuat IAM role untuk interaksi lintas layanan .....	30
Mempersiapkan game untuk FleetIQ .....	37
Langkah-langkah integrasi .....	37
Mengelola grup server game .....	39
Membuat grup server game .....	40
Memperbarui grup server game .....	41
Melacak instans grup server game .....	41
Mengintegrasikan server game .....	41
Mendaftarkan server game .....	41
Memperbarui status server game .....	42
Keluar dari pendaftaran server game .....	43
Integrasikan Client Game .....	43
Biarkan Amazon GameLift FleetIQ memilih server game .....	44
Pilih server game Anda sendiri .....	44
Monitor dengan CloudWatch .....	46
Referensi Amazon GameLift FleetIQ .....	49

---

Referensi API layanan (SDK AWS) .....	49
Tindakan API Amazon GameLift FleetIQ .....	49
Bahasa pemrograman yang tersedia .....	50
Keamanan dengan FleetIQ .....	52
Catatan rilis dan versi SDK .....	53
Semua GameLift panduan Amazon .....	54
AWSGlosarium .....	55
.....	lvi

# Apa itu Amazon GameLift FleetIQ?

Amazon GameLift FleetIQ mengoptimalkan penggunaan Instans Spot Amazon Elastic Compute Cloud (Amazon EC2) berbiaya rendah untuk hosting game berbasis cloud. Dengan Amazon GameLift FleetIQ, Anda dapat bekerja secara langsung dengan sumber daya hosting Anda di Amazon EC2 dan Amazon EC2 Auto Scaling sambil memanfaatkan GameLift optimasi Amazon untuk menghadirkan hosting game yang murah dan tangguh untuk pemain Anda. Instans Spot Amazon EC2, meskipun ditawarkan dengan diskon besar, umumnya tidak layak untuk hosting game karena ketersediaannya tidak menentu dan ada potensi [gangguan](#). Amazon GameLift FleetIQ secara signifikan mengurangi batasan ini, membuat penggunaan Instans Spot berbiaya rendah layak untuk hosting game.

Optimalisasi FleetIQ juga tersedia saat menggunakan Amazon GameLift untuk mengelola hosting game Anda. Untuk informasi tentang opsi GameLift hosting Amazon, lihat [Panduan GameLift Pengembang Amazon](#).

Solusi hosting game Amazon GameLift FleetIQ dirancang untuk pengembang game yang:

- Memiliki AWS penerapan yang ada atau ingin menggunakan Amazon EC2 secara langsung daripada melalui layanan Amazon yang dikelola sepenuhnya. GameLift Amazon GameLift FleetIQ bekerja dengan grup Penskalaan Otomatis EC2 yang Anda kelola Akun AWS, memberi Anda akses penuh ke instans dan grup EC2 Anda. Anda juga dapat berintegrasi dengan AWS layanan lain, termasuk Amazon Elastic Container Service (Amazon ECS), Amazon Elastic Kubernetes Service (Amazon EKS), dan AWS Shield Advanced
- Memiliki hosting game on-premises yang ada dan ingin memperluas kapasitas ke cloud. Dengan Amazon GameLift FleetIQ, Anda dapat membangun sistem penerapan hibrid yang menggunakan kapasitas lokal Anda dan secara bertahap menambah AWS kapasitas cloud sesuai kebutuhan.

## Siap untuk mulai bekerja dengan Amazon GameLift FleetIQ?

- Pelajari cara menggunakan Amazon GameLift FleetIQ untuk game Anda dengan mengikuti kursus [Menggunakan Amazon Amazon GameLift FleetIQ untuk Server Game](#) di Skill Builder. AWS Untuk ikhtisar kursus terkait, lihat [Rencana Pembelajaran Teknologi Game](#). Beberapa kursus tersedia dalam berbagai bahasa.
- Ikuti instruksi di [Langkah-langkah integrasi Amazon GameLift FleetIQ](#).

# Cara kerja Amazon GameLift FleetIQ

Solusi Amazon GameLift FleetIQ adalah lapisan hosting game yang melengkapi set lengkap alat manajemen sumber daya komputasi yang Anda dapatkan dengan Amazon EC2 dan Auto Scaling. Selain menawarkan serangkaian fitur khusus untuk hosting game, Amazon GameLift FleetIQ menyediakan lapisan logika tambahan yang memungkinkan untuk menggunakan Instans Spot berbiaya rendah untuk hosting game. Solusi ini memungkinkan Anda langsung mengelola sumber daya Amazon EC2 dan Auto Scaling dan mengintegrasikan sesuai kebutuhan dengan layanan AWS .

Saat menggunakan Amazon GameLift FleetIQ, Anda bersiap untuk meluncurkan instans Amazon EC2 seperti biasa: buat Amazon Machine Image (AMI) dengan perangkat lunak server game Anda, buat template peluncuran Amazon EC2, dan tentukan pengaturan konfigurasi untuk grup Auto Scaling. Namun, alih-alih membuat grup Auto Scaling secara langsung, Anda membuat grup server game Amazon FleetIQ dengan sumber GameLift daya dan konfigurasi Amazon EC2 dan Auto Scaling Anda. Tindakan ini meminta Amazon GameLift FleetIQ untuk membuat grup server game dan grup Auto Scaling yang sesuai. Grup server game terkait dengan dan mengelola aspek-aspek tertentu dari grup Auto Scaling.

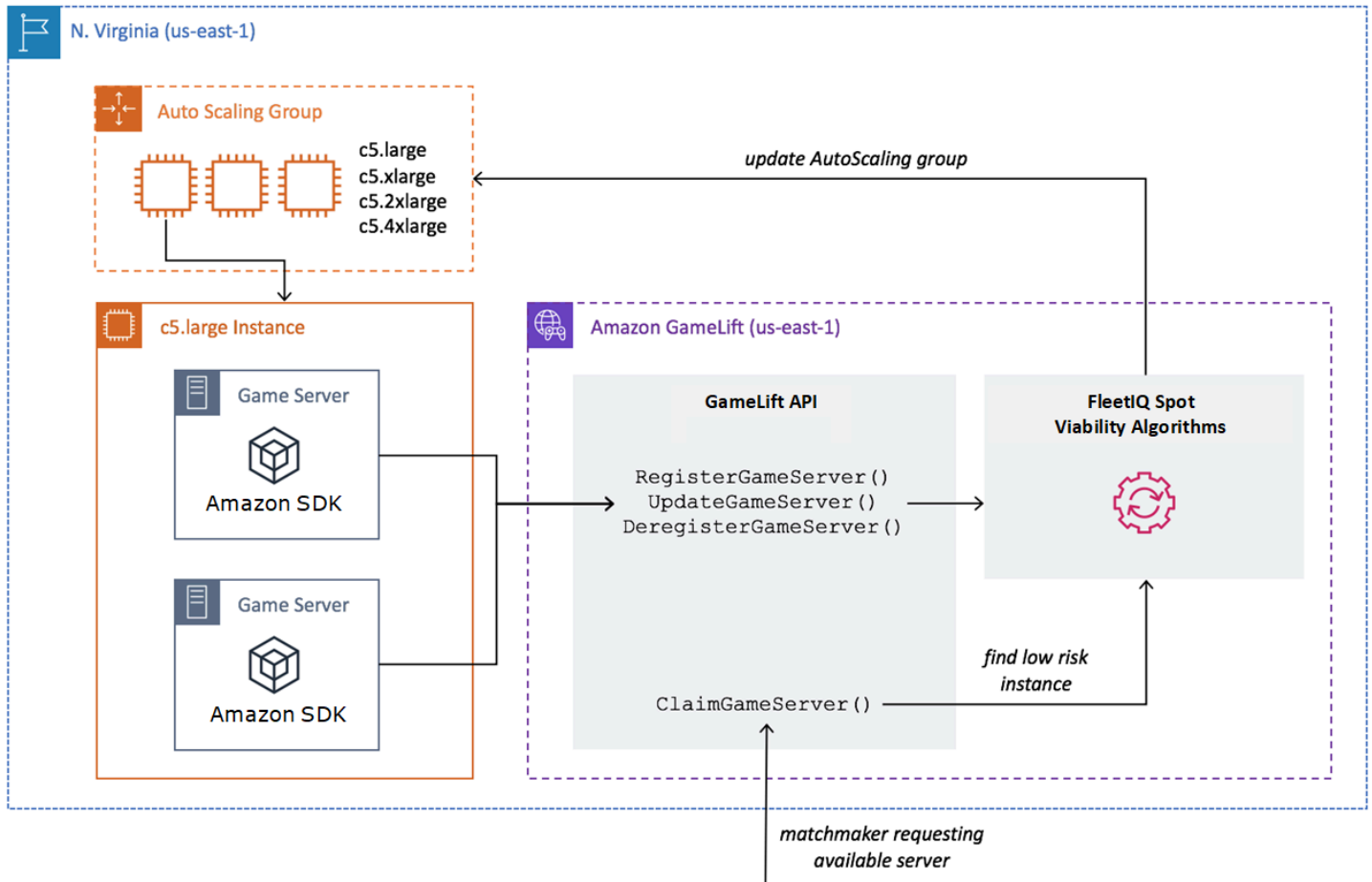
Setelah grup Auto Scaling dibuat, Anda memiliki akses penuh ke sumber daya Amazon EC2 dan Auto Scaling. Anda dapat mengubah konfigurasi grup Auto Scaling, menambahkan kebijakan penskalaan multi-level atau penyeimbang beban, dan berintegrasi dengan layanan lain. AWS Anda dapat terhubung langsung ke instans dalam grup. Sebagai bagian dari logika pengoptimalannya, Amazon GameLift FleetIQ juga membuat pembaruan berkala ke properti grup Auto Scaling tertentu. Anda dapat melacak status ketersediaan semua instans yang di-deploy oleh grup Auto Scaling.

Anda dapat menangguhkan sementara aktivitas Amazon GameLift FleetIQ untuk grup server game kapan saja. Anda juga memiliki pilihan untuk menghapus grup server game namun mempertahankan grup Auto Scaling yang sesuai.

## Logika Amazon GameLift FleetIQ

Diagram berikut menggambarkan peran Amazon GameLift FleetIQ saat bekerja dengan Amazon EC2 untuk hosting game. Tujuan utamanya adalah untuk menemukan server game terbaik untuk menjadi host sesi game dan memberikan pemain pengalaman gameplay yang optimal. Amazon GameLift FleetIQ mendefinisikan sumber daya terbaik sebagai sumber daya yang memberikan viabilitas hosting game tertinggi dengan biaya terendah. Amazon GameLift FleetIQ mendekati tujuan ini dengan dua cara utama: pertama dengan hanya mengizinkan jenis instans yang layak di grup

Auto Scaling, dan kedua dengan menempatkan sesi permainan baru secara efektif di seluruh sumber daya grup yang tersedia.



## Isi grup Auto Scaling dengan jenis instans yang optimal

Tugas grup Auto Scaling adalah meluncurkan instans baru dan mempersiapkan instans lama, mempertahankan koleksi sumber daya hosting dan menskalakannya untuk memenuhi permintaan pemain Anda. Untuk melakukannya, grup Auto Scaling bergantung pada daftar jenis instans yang Anda inginkan. Tugas Amazon GameLift FleetIQ adalah untuk terus memeriksa kelayakan jenis instans yang diinginkan ini dan memperbarui daftar untuk grup Auto Scaling. Proses ini disebut penyeimbangan instans. Ini memastikan bahwa instans dalam grup Auto Scaling terus menerus disegarkan sehingga hanya jenis instans yang layak saat ini yang digunakan setiap saat.

Amazon GameLift FleetIQ memengaruhi cara grup Auto Scaling memilih jenis instans optimal dengan cara berikut:

- Ini menentukan penggunaan Instans Spot dan/atau On-Demand. Grup server game Amazon GameLift FleetIQ dikonfigurasi dengan strategi penyeimbangan, yang memengaruhi cara grup Auto

Scaling menggunakan Instans Spot dan/atau Sesuai Permintaan. Instans Spot memiliki biaya lebih rendah karena ketersediaan yang berfluktuasi dan potensi [gangguan](#), batasan yang diminimalkan Amazon FleetIQ GameLift untuk hosting server game. Instans Sesuai Permintaan lebih mahal namun menawarkan ketersediaan yang lebih andal ketika Anda membutuhkannya.

- Ini membatasi instance baru untuk diluncurkan hanya pada jenis instance yang layak. Grup server game Amazon GameLift FleetIQ mengelola daftar master jenis instans yang Anda inginkan, Proses penyeimbangan instans terus-menerus mengevaluasi setiap jenis instans yang diinginkan dalam daftar untuk kelangsungan hidup hosting game, menggunakan algoritme prediksi yang melihat ketersediaan dan tingkat interupsi tipe instans terbaru. Sebagai hasil dari evaluasi ini, Amazon GameLift FleetIQ terus memperbarui daftar tipe instans yang diinginkan grup Auto Scaling untuk menyertakan hanya jenis instans yang layak saat ini.
- Ini menandai instance yang ada yang merupakan tipe instance yang tidak layak. Amazon GameLift FleetIQ mengidentifikasi instans yang ada di grup Auto Scaling yang saat ini merupakan tipe instans yang tidak layak. Instans ini ditandai sebagai mengering, yang artinya instans dihentikan dan diganti dengan instans baru. Untuk instans yang memiliki perlindungan server game diaktifkan, penghentian ditunda sampai sesi game aktif berakhir secara normal.

Saat grup Auto Scaling meluncurkan dan menghentikan instans, grup ini mempertahankan koleksi yang dioptimalkan untuk hosting game meskipun ketersediaan jenis Instans Spot berbiaya rendah berfluktuasi. Aktivitas penyeimbangan berlangsung pada grup server game dengan instans aktif saja. Pelajari selengkapnya tentang cara kerja proses ini di [Proses penyeimbangan spot](#).

## Tempatkan sesi game secara efektif

Amazon GameLift FleetIQ melacak semua server game aktif di grup server game dan menggunakan informasi ini untuk menentukan penempatan terbaik untuk sesi dan pemain game baru.

Untuk mengaktifkan Amazon GameLift FleetIQ melacak server game, perangkat lunak server game Anda harus melaporkan statusnya. AMI kustom Anda mengontrol bagaimana proses server game baru dimulai dan berhenti pada setiap instans. Ketika server game baru dimulai, ia mendaftar dengan Amazon GameLift FleetIQ, menunjukkan bahwa ia siap untuk menyelenggarakan sesi permainan. Setelah terdaftar, server game secara berkala melaporkan kondisinya dan apakah saat ini menjadi host sesi game. Ketika server game dimatikan, itu tidak mendaftar dengan Amazon FleetIQ. GameLift

Untuk memulai sesi permainan baru, klien game Anda (atau mak comblang atau layanan klien lainnya) mengirimkan permintaan server game ke Amazon GameLift FleetIQ. Amazon GameLift FleetIQ menemukan server game yang tersedia, mengklaimnya untuk sesi game baru, dan



merespons dengan ID server game dan info koneksi. Game Anda kemudian meminta server game untuk memperbarui statusnya dan memulai sesi game baru untuk pemain yang masuk.

Saat memilih server game untuk menyelenggarakan sesi game baru, Amazon GameLift FleetIQ menggunakan proses pengambilan keputusan berikut untuk mengoptimalkan penempatan dengan Instans Spot berbiaya rendah yang layak:

1. Jika memungkinkan, Amazon GameLift FleetIQ menempatkan sesi permainan baru pada instance yang sudah menjadi tuan rumah sesi game lainnya. Dengan mengemas (tapi tidak terlalu banyak) beberapa instans dan menjaga yang lain siaga, grup Auto Scaling dapat dengan cepat menurunkan skala instans siaga saat tidak diperlukan, yang dapat menurunkan biaya hosting.
2. Amazon GameLift FleetIQ mengabaikan instance yang ditandai sebagai menguras, yaitu, tidak layak untuk hosting game. Instans ini terus berjalan hanya untuk mendukung sesi game yang ada. Mereka tidak dapat digunakan untuk sesi game baru kecuali tidak ada server game lain yang tersedia.
3. Amazon GameLift FleetIQ mengidentifikasi semua server game yang tersedia yang berjalan pada instance yang layak.

Anda dapat mengaktifkan perlindungan sesi game untuk grup server game untuk mencegah grup Auto Scaling menghentikan instans dengan sesi game yang aktif berjalan.

## Sumber daya dan komponen utama

Buat sumber daya berikut di AWS akun Anda sebelum menyiapkan sumber daya hosting game dengan Amazon GameLift FleetIQ. Sebagai praktik terbaik, kembangkan dan uji deployment server game Anda dengan sumber daya ini sebelum menggunakannya melalui grup server game.

- Gambar Mesin Amazon (AMI). AMI adalah templat untuk konfigurasi perangkat lunak tertentu yang ingin Anda luncurkan dengan instans Amazon EC2 Anda. Untuk hosting game, AMI Anda mencakup sistem operasi, biner atau kontainer server game Anda, dan perangkat lunak waktu aktif lainnya yang dibutuhkan server game Anda. Untuk informasi selengkapnya tentang membuat AMI, lihat [Gambar Mesin Amazon](#) di Panduan Pengguna Amazon EC2. AMI bersifat khusus Wilayah. Anda dapat menyalin AMI dari satu Wilayah ke Wilayah lainnya, seperti yang dijelaskan dalam [Menyalin AMI](#) di Panduan Pengguna Amazon EC2.
- Templat peluncuran Amazon EC2. Templat peluncuran menyediakan petunjuk untuk menjalankan dan mengelola instans dalam grup Auto Scaling. Ini menentukan AMI, menyediakan daftar jenis instans yang sesuai, dan mengatur jaringan, keamanan, dan properti lainnya. Untuk informasi

selengkapnya tentang templat peluncuran, lihat bagian [Meluncurkan Instans dari Templat Peluncuran](#) dalam Panduan Pengguna Amazon EC2. Templat peluncuran bersifat khusus Wilayah.

- **AWS Peran IAM.** Peran IAM mendefinisikan sekumpulan izin yang memungkinkan akses terbatas ke sumber daya. AWS Entitas tepercaya, seperti AWS layanan lain, dapat mengambil peran dan mewarisi izinnya. Saat menggunakan Amazon GameLift FleetIQ, Anda harus memberikan peran IAM dengan kebijakan terkelola yang memungkinkan Amazon GameLift FleetIQ membuat dan mengakses grup Auto Scaling dan sumber daya instans EC2 di akun Anda. AWS IAM role tidak bersifat khusus Wilayah.

Amazon GameLift FleetIQ mengelola sumber daya berikut secara langsung dan memiliki otoritas langsung atas sumber daya tersebut.

- **GameLift grup server game.** Grup server game berisi pengaturan konfigurasi yang menentukan cara kerja Amazon GameLift FleetIQ dengan grup Auto Scaling yang sesuai untuk menghadirkan hosting game berbiaya rendah. Grup server game bersifat khusus Wilayah. Saat Anda membuat grup server game di Wilayah, grup Auto Scaling baru secara otomatis dibuat di AWS akun Anda di Wilayah yang sama. Grup server game terhubung ke grup Auto Scaling dan memiliki akses (dengan mengambil IAM role) untuk mengelola dan memodifikasi beberapa pengaturannya. Grup server game adalah sumber daya berumur panjang; developer sebaiknya tidak terlalu sering membuatnya. Grup server game juga merupakan sumber pengelompokan fungsional untuk server game yang di-host pada instance di grup Auto Scaling dan terdaftar di Amazon FleetIQ. GameLift
- **GameLift server permainan.** Sumber daya server game mewakili eksekusi game yang berjalan pada instance yang terkait dengan grup server game Amazon GameLift FleetIQ. Sumber daya ini dibuat ketika server game mendaftar dengan Amazon GameLift FleetIQ dan mengidentifikasi grup server game yang dimilikinya. Amazon GameLift FleetIQ melacak status pemanfaatan dan status klaim dari setiap server game terdaftar, yang memungkinkannya memantau ketersediaan server game. Server game bersifat khusus Wilayah, yang artinya mereka terkait dengan grup server game khusus Wilayah. Ketika game Anda meminta server game baru, ia menentukan grup server game dan Wilayah.

Sumber daya ini dibuat melalui sumber daya Amazon GameLift FleetIQ. Sumber daya ini diciptakan di akun AWS Anda dan Anda memiliki kontrol penuh.

- **Grup Auto Scaling Amazon EC2.** Grup Auto Scaling meluncurkan dan mengelola kumpulan instans EC2, dan secara otomatis menskalakan kapasitas grup. Dengan Amazon GameLift FleetIQ, ada

one-to-one hubungan antara grup server game dan grup Auto Scaling. Meskipun Anda dapat memperbarui semua pengaturan untuk grup Auto Scaling, Amazon GameLift FleetIQ secara berkala mengganti dan memperbarui pengaturan tertentu sebagai bagian dari logikanya untuk menyeimbangkan Instans Spot untuk kelayakan hosting game. Untuk informasi selengkapnya, lihat [AutoScalingGroup](#) di Panduan Pengguna Auto Scaling Amazon EC2. Grup Auto Scaling bersifat khusus Wilayah; dibuat di Wilayah yang sama dengan grup server game.

- Instans Amazon EC2. Sebuah instans adalah server virtual di cloud. Jenis instans memiliki konfigurasi perangkat keras tertentu yang menentukan komputasi, memori, disk, dan sumber daya jaringan. Jenis instans biasanya diluncurkan oleh grup Auto Scaling dengan AMI. Instans dapat berupa Spot atau Sesuai Permintaan, tergantung pada ketersediaan. Dengan Amazon GameLift FleetIQ, instans menjalankan satu atau beberapa proses server game, yang masing-masing dapat menghosting beberapa sesi game. Instans bersifat khusus wilayah karena terkait dengan grup Auto Scaling khusus Wilayah.

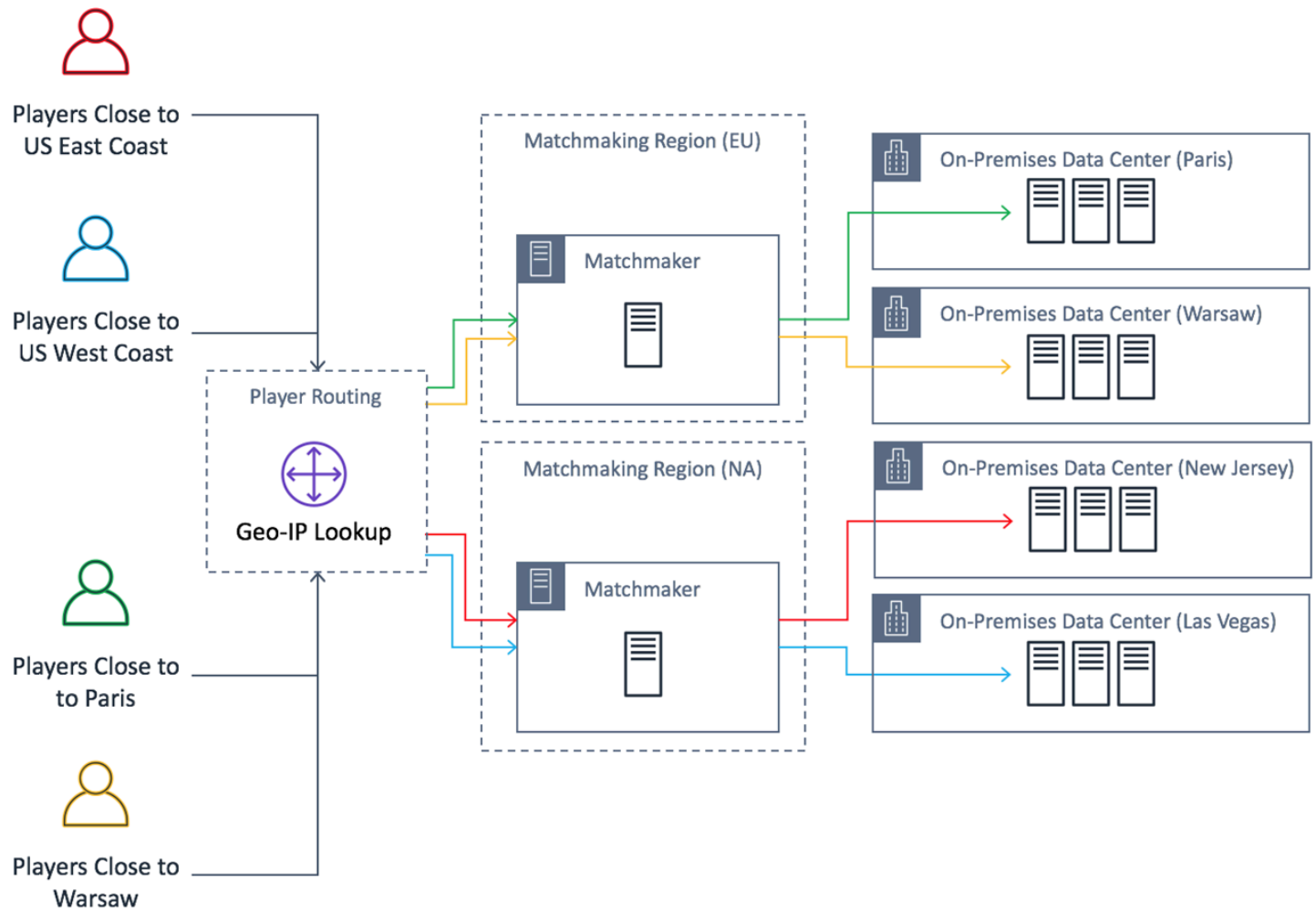
## Arsitektur game dengan Amazon GameLift FleetIQ

### Melengkapi hosting on-premises

Amazon GameLift FleetIQ dirancang untuk menggunakan kembali backend game Anda yang ada, termasuk perutean Geo-IP pemain, perjodohan, atau layanan lobi yang mungkin sudah Anda miliki. Contoh berikut menggambarkan bagaimana Amazon GameLift FleetIQ mungkin cocok dengan penerapan lokal yang ada.

#### Example

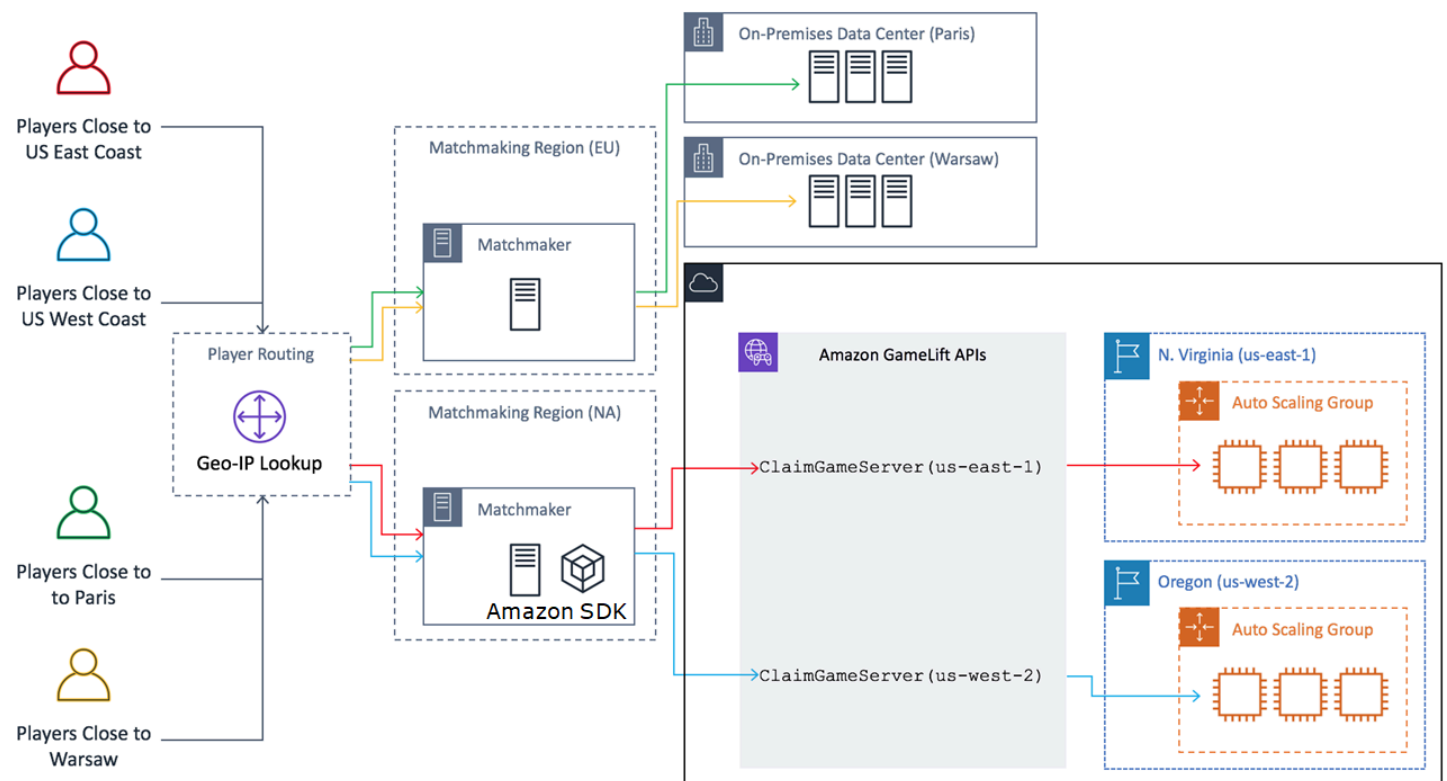
Dalam contoh ini, hosting game awalnya ditangani dengan empat pusat data eksklusif untuk menjadi host pemain di Amerika Utara dan Eropa. Tergantung pada perkiraan lokasi fisik mereka, pemain diarahkan ke salah satu dari dua matchmaker wilayah. Matchmaker mengelompokkan pemain berdasarkan keterampilan dan latensi dan kemudian menempatkan mereka ke server game terdekat untuk meminimalkan lag.



Pengembang game ingin mengganti server game Amerika Utara mereka dengan server yang disediakan oleh Amazon GameLift FleetIQ. Untuk memulai, mereka membuat pembaruan kecil ke server game mereka untuk mengaktifkannya digunakan dengan Amazon GameLift FleetIQ dan kemudian membuat Amazon Machine Image (AMI). Citra ini akan diinstal pada setiap instans EC2 yang di-deploy untuk game. Citra berisi server game, dependensi, dan hal lain yang diperlukan untuk menjalankan sesi game untuk pemain.

Dengan siap AMI, pengembang membuat dua grup server game Amazon GameLift FleetIQ, satu untuk setiap Wilayah Amerika AWS Utara (`us-east-1` dan `us-west-2`) Pengembang meneruskan template peluncuran (yang menyediakan AMI), daftar jenis instans yang diinginkan, dan pengaturan konfigurasi lainnya untuk grup. Daftar jenis instans yang diinginkan memberi tahu Amazon GameLift FleetIQ jenis mana yang akan digunakan saat memeriksa Instans Spot yang layak untuk hosting game.

Akhirnya, pengembang mengintegrasikan AWS SDK dengan Amazon GameLift FleetIQ ke dalam mak comblang Amerika Utara mereka, yang memanggil Amazon GameLift FleetIQ ketika sekelompok pemain baru membutuhkan kapasitas server untuk sesi permainan. Amazon GameLift FleetIQ menempatkan Instans Spot dengan server game yang tersedia, menyimpannya untuk pemain, dan menyediakan informasi koneksi server. Pemain terhubung ke server, memainkan game, dan memutuskan sambungan. Untuk memulai permainan baru, pemain masuk kembali perjudohan, yang meminta Amazon GameLift FleetIQ untuk menemukan server game lain yang tersedia. Setiap permintaan game baru memicu Amazon GameLift FleetIQ untuk mencari dan memilih server game dengan kemungkinan gangguan yang rendah. Akibatnya, Amazon GameLift FleetIQ terus-menerus mengalihkan pemain dari server game yang tidak layak untuk hosting game, bahkan saat ketersediaan Instans Spot berfluktuasi seiring waktu.



## Siklus hidup dan status Amazon GameLift FleetIQ

### Proses penyeimbangan spot

Amazon GameLift FleetIQ secara berkala menyeimbangkan instans dalam grup Penskalaan Otomatis yang memiliki Instans Spot. Proses ini tidak aktif dengan grup-grup server game yang menggunakan strategi penyeimbangan ON\_DEMAND\_ONLY atau tidak memiliki instans aktif.

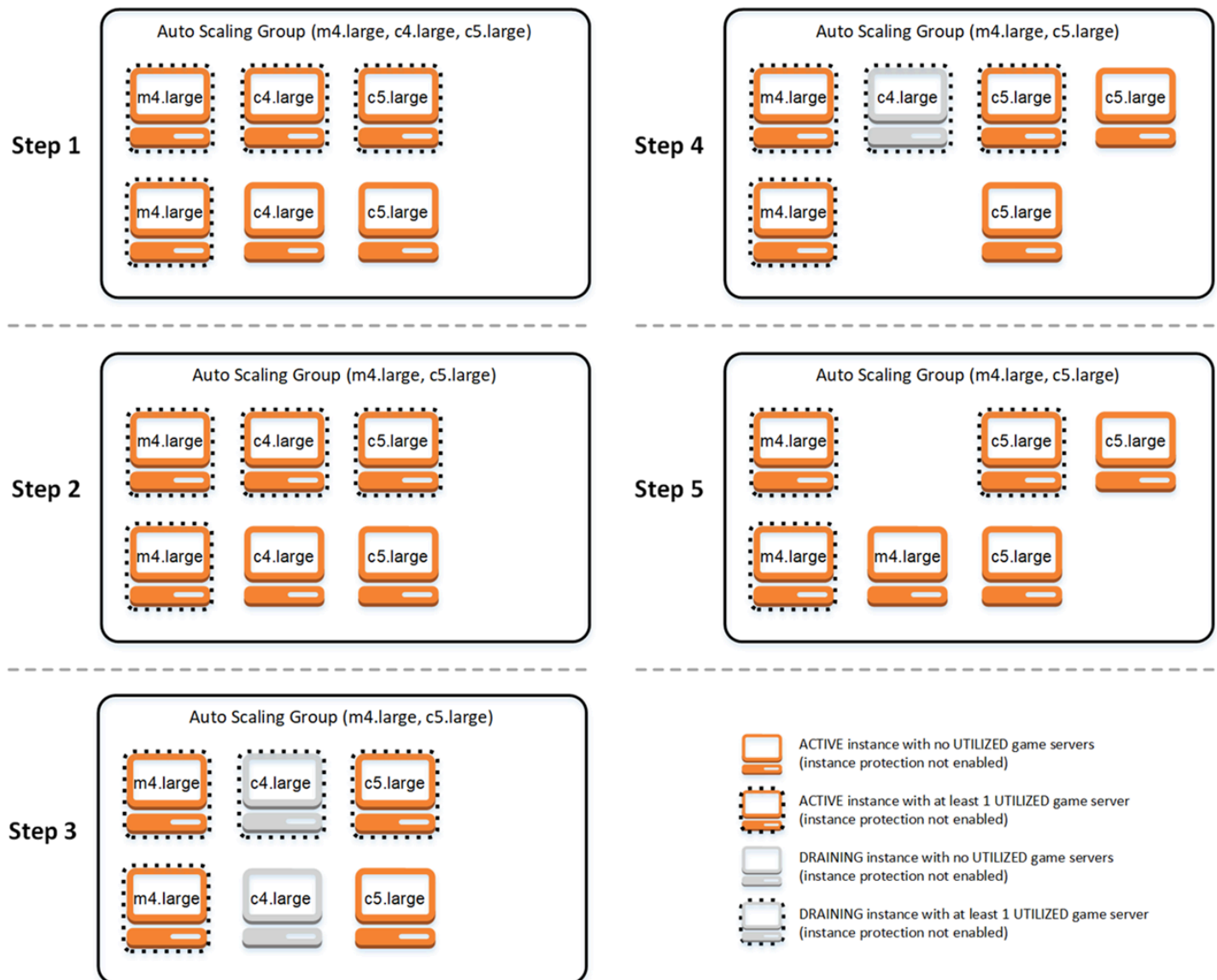
Penyeimbangan spot memiliki dua tujuan utama:

- Untuk terus-menerus menyegarkan grup dengan hanya menggunakan jenis Instans Spot yang layak untuk hosting game.
- Untuk menggunakan beberapa jenis instans layak (jika mungkin) untuk mengurangi dampak dari gangguan server game tak terduga.

Amazon GameLift FleetIQ menyeimbangkan dengan mengevaluasi jenis instans grup dan menghapus instans yang lebih mungkin mengakibatkan gangguan server game. Untuk terhindar dari menghentikan kejadian dalam gameplay aktif selama penyeimbangan, praktik terbaik adalah mengaktifkan perlindungan server game untuk grup server game yang sedang diproduksi.

### Example

Contoh berikut menggambarkan bagaimana instans dalam grup Auto Scaling dipengaruhi oleh penyeimbangan Spot.



- Langkah 1. Melalui grup server game, grup Auto Scaling terkait diatur untuk meluncurkan instans jenis m4.large, c4.large, dan c5.large dengan perlindungan server game diaktifkan. Grup Auto Scaling telah meluncurkan koleksi seimbang yang terdiri dari dua Instans Spot dari masing-masing jenis. Empat instans memiliki setidaknya satu server game dalam status UTILIZED (ditampilkan dengan border putus-putus), sementara dua instans saat ini tidak mendukung gameplay.
- Langkah 2. Amazon GameLift FleetIQ mengevaluasi kelayakan hosting game saat ini dari ketiga jenis instans. Evaluasi menentukan bahwa jenis instans c4.large memiliki potensi yang tidak dapat diterima untuk gangguan server game. Amazon GameLift FleetIQ segera memperbarui konfigurasi grup Auto Scaling untuk sementara menghapus c4.large dari daftar jenis instans, mencegah instans c4.large tambahan diluncurkan.

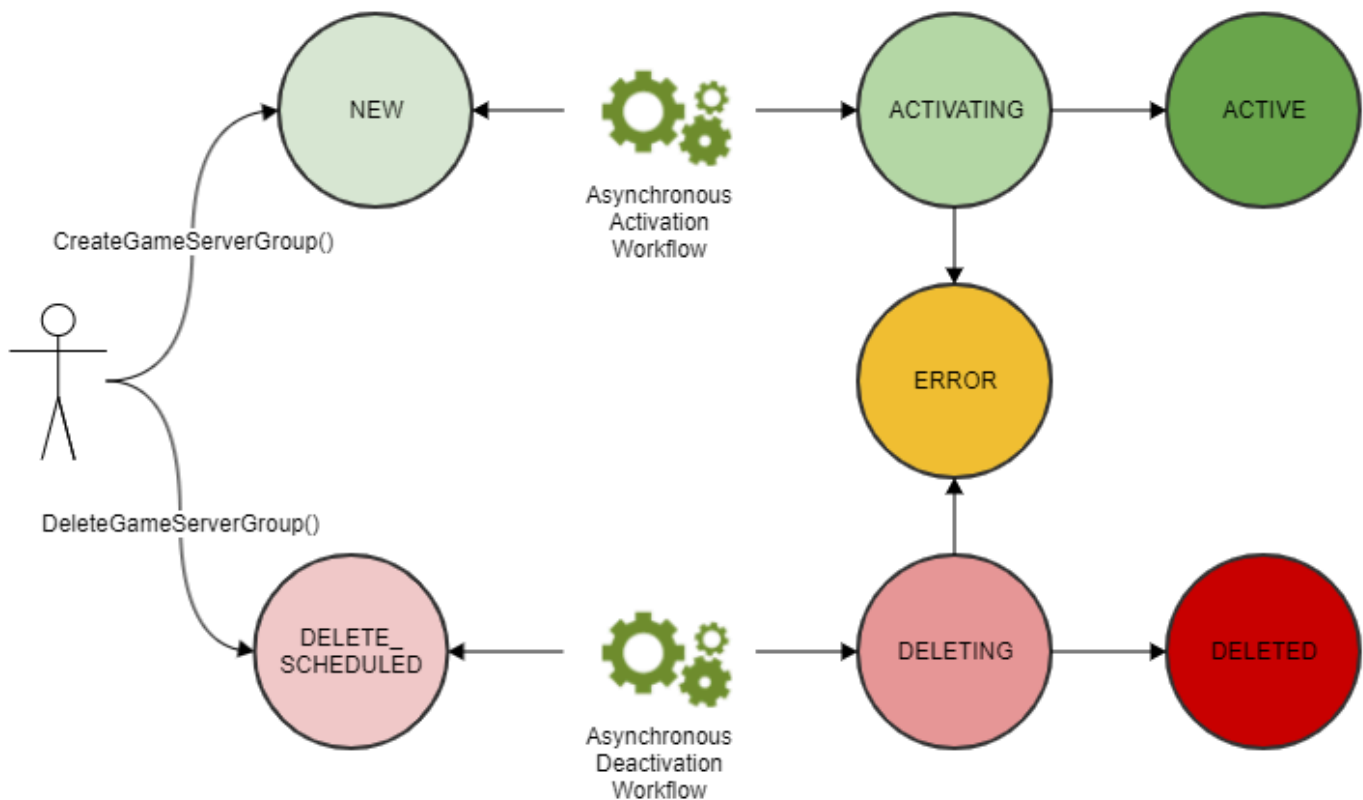
- Langkah 3. Amazon GameLift FleetIQ mengidentifikasi instans tipe c4.large yang ada dan mengambil tindakan untuk menghapusnya dari grup. Sebagai langkah pertama, semua server game yang berjalan pada instans c4.large ditandai sebagai draining. Server game pada instans yang draining dapat diklaim hanya sebagai upaya terakhir jika tidak ada server game lain yang tersedia. Selain itu, grup Auto Scaling dengan instans draining dipicu untuk meluncurkan instans baru untuk menggantikannya.
- Langkah 4. Saat instans baru yang layak menjadi online, grup Auto Scaling akan menghentikan instans draining. Penggantian ini memastikan bahwa kapasitas yang diinginkan grup dipertahankan. Instans pertama yang akan dihentikan adalah instans c4.large tanpa server game dimanfaatkan dan perlindungan server game dimatikan. Ini diganti dengan instans c5.large baru.
- Langkah 5. Instans draining dengan perlindungan server game terus berjalan sementara server game mereka mendukung gameplay. Ketika gameplay berakhir, instans c4.large yang tersisa dihentikan ketika sebuah instans m4.large baru telah diluncurkan untuk mengambil alih.

Sebagai hasil dari proses ini, grup Auto Scaling mempertahankan kapasitas yang diinginkan sementara grup menyeimbangkan dari menggunakan tiga jenis instans menjadi dua jenis instans. Amazon GameLift FleetIQ terus mengevaluasi daftar asli jenis instans untuk kelayakan hosting game. Ketika c4.large kembali dianggap sebagai jenis instans yang layak, grup Auto Scaling diperbarui untuk mencakup semua tiga jenis instans. Grup ini secara alami menyeimbangkan dari waktu ke waktu.

## Kehidupan grup server game

Grup server game melalui siklus hidup berikut, termasuk penyediaan dan pembaruan status. Grup server game diharapkan menjadi sumber daya berumur panjang.





- Anda membuat grup server game dengan memanggil GameLift API Amazon `CreateGameServerGroup()` dan meneruskan templat peluncuran EC2 dan pengaturan konfigurasi. Menanggapi panggilan tersebut, grup server game baru dibuat dan ditempatkan dalam status `NEW`.
- Amazon GameLift FleetIQ mengaktifkan alur kerja aktivasi asinkron, mengalihkan status grup server game ke `MENGAKTIFKAN`. Alur kerja memulai penciptaan sumber daya yang mendasari, termasuk grup Amazon EC2 Auto Scaling dan instans EC2 dengan AMI disediakan.
- Jika penyediaan gagal karena alasan apapun, grup server game ditempatkan ke status `ERROR`. Untuk mendapatkan informasi kesalahan tambahan untuk membantu debug penyebab kegagalan, panggil `DescribeGameServerGroup()` pada grup server game yang mengalami keadaan kesalahan.
- Jika penyediaan berhasil, grup server game dialihkan ke status `ACTIVE`. Pada titik ini, instance diluncurkan dengan server game yang mendaftar dengan Amazon GameLift FleetIQ. Jenis instans grup secara berkala dievaluasi untuk kelayakan hosting game dan diseimbangkan sesuai kebutuhan. Amazon GameLift FleetIQ juga melacak status server game aktif dalam grup dan merespons permintaan server game.

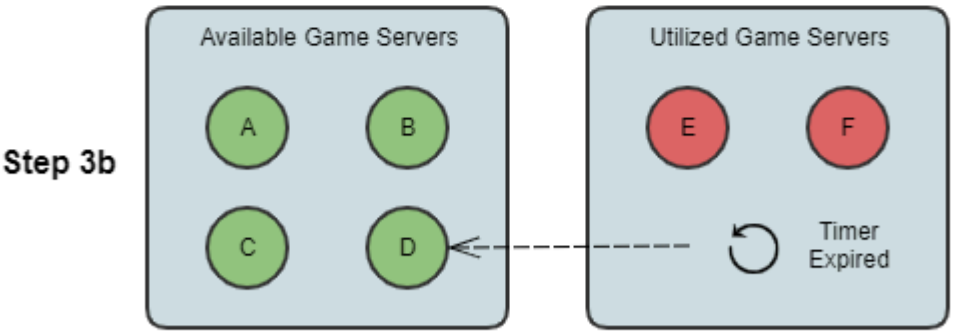
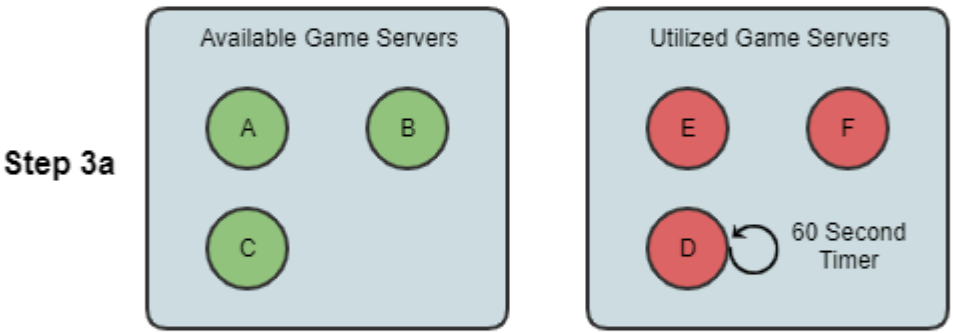
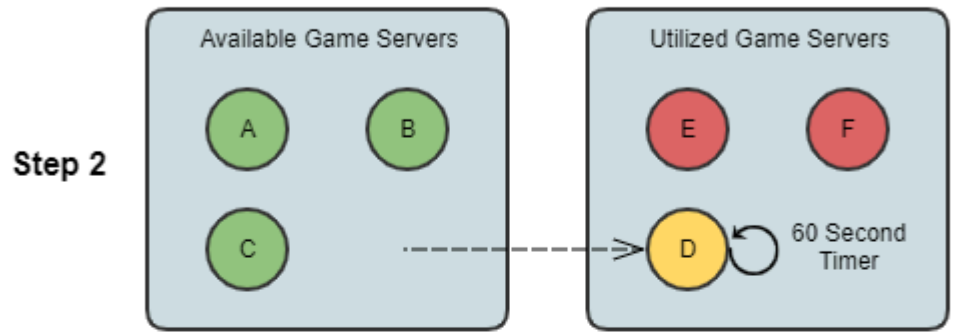
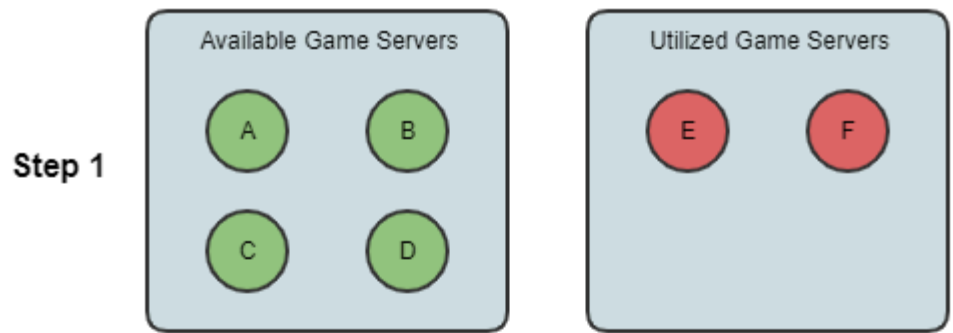
- Anda menghapus grup server game dengan memanggil `DeleteGameServerGroup()` dengan pengidentifikasi grup. Tindakan ini menempatkan grup server game ke dalam status `DELETE_SCHEDULED`. Hanya grup server game dalam keadaan `ACTIVE` atau `ERROR` yang dapat dijadwalkan untuk dihapus.
- Amazon GameLift FleetIQ mengaktifkan alur kerja penonaktifan asinkron sebagai respons terhadap status `DELETE_SCHEDULED`, mengalihkan status grup server game ke `DELETING`. Anda memiliki pilihan untuk menghapus hanya grup server game atau menghapus grup server game dan grup Auto Scaling terkait.
  - Jika penyediaan gagal karena alasan apapun, grup server game ditempatkan ke status `ERROR`. Untuk mendapatkan informasi kesalahan tambahan untuk membantu debug penyebab kegagalan, panggil `DescribeGameServerGroup()` pada grup server game yang mengalami keadaan kesalahan.
  - Jika penonaktifan berhasil, grup server game dialihkan ke status `DELETED`.

## Kehidupan server game

Dengan Amazon GameLift FleetIQ, server game menjalani siklus hidup berikut, termasuk penyediaan dan pembaruan status. Sebuah server game diharapkan menjadi sumber daya berumur pendek. Sebagai praktik terbaik, server game harus keluar dari daftar setelah akhir sesi game ketimbang digunakan kembali untuk sesi game lainnya. Pendekatan ini membantu memastikan bahwa server game yang tersedia selalu berjalan pada sumber daya berbiaya terendah yang layak untuk game hosting.

- Sumber daya server game dibuat saat proses server game, berjalan pada instans dalam grup Penskalaan Otomatis GameLift Tertaut Amazon FleetIQ, memanggil Amazon GameLift API `RegisterGameServer()` untuk memberi tahu Amazon GameLift FleetIQ bahwa ia siap menjadi tuan rumah pemain dan gameplay. Sebuah server game memiliki dua status untuk melacak ketersediaan saat ini:
  - Status pemanfaatan melacak apakah server game saat ini mendukung gameplay. Status ini awalnya diatur ke `AVAILABLE`, menunjukkan bahwa ia siap untuk menerima gameplay baru. Setelah server game ditempati oleh gameplay, status ini diatur ke `UTILIZED`.
  - Status klaim melacak apakah server game diklaim untuk gameplay dalam waktu dekat. Sebuah server game dalam status `CLAIMED` menunjukkan bahwa itu telah sementara direservasi oleh client game (atau layanan game seperti matchmaker). Status ini mencegah Amazon GameLift FleetIQ menyediakan server game yang sama kepada beberapa pemohon. Server game dengan status klaim kosong tersedia untuk diklaim.

- Diagram berikut menggambarkan bagaimana status pemanfaatan server game dan status klaim berubah selama rentang hidupnya.



-  Utilization Status is AVAILABLE, no Claim Status
-  Utilization Status is AVAILABLE, Claim Status is CLAIMED
-  Utilization Status is UTILIZED, Claim Status can be either

- Langkah 1. Sebuah grup server game memiliki enam server game terdaftar. Empat memiliki status pemanfaatan AVAILABLE (A, B, C, dan D), dan dua saat ini UTILIZED (E dan F).
- Langkah 2. Klien game atau sistem perjodohan memanggil Amazon GameLift API `ClaimGameServer()` untuk meminta server game baru. Permintaan ini meminta Amazon GameLift FleetIQ untuk mencari server game (D) yang tersedia dan menetapkan status klaimnya ke CLAIM selama 60 detik. Amazon GameLift FleetIQ menanggapi permintaannya dengan informasi koneksi untuk server game (alamat IP dan port), serta data spesifik game opsional lainnya. Karena gameplay belum dimulai pada server game, status pemanfaatannya tetap AVAILABLE, tetapi tidak dapat diklaim oleh permintaan lain.
- Langkah 3a. Dengan menggunakan informasi koneksi yang disediakan, client game dapat terhubung ke server game dan memulai gameplay. Server game (D) harus dipicu dalam waktu 60 detik untuk mengubah status pemanfaatannya menjadi UTILIZED dengan memanggil Amazon API `GameLift UpdateGameServer()`
- Langkah 3b. Jika status pemanfaatan server game tidak diperbarui dalam waktu 60 detik, timer klaim kedaluwarsa dan status klaim diatur ulang ke kosong. Server game (D) dikembalikan ke kolom server game yang tersedia dan tidak diklaim.
- Sumber daya server game dihapus setelah gameplay di server game selesai dan pemain telah putus sambungan. Sebelum mematikan, proses server game memanggil Amazon GameLift API `DeregisterGameServer()` untuk memberi tahu Amazon GameLift FleetIQ tentang keberangkatannya dari kumpulan server game grup game.

## Praktik terbaik Amazon GameLift FleetIQ

Amazon GameLift FleetIQ adalah lapisan logika tingkat rendah yang membantu Anda mengelola sumber daya Amazon EC2 untuk hosting game. Secara khusus, Amazon GameLift FleetIQ mengoptimalkan penggunaan Instans Spot yang layak untuk hosting game dengan meminimalkan kemungkinan sesi game mungkin terganggu. Ini juga menyediakan fungsi hosting game dasar untuk melacak server game yang tersedia dan merute gameplay ke server game berbiaya rendah dan kemampuan tinggi.

Amazon GameLift FleetIQ sebagai fitur mandiri tidak menyediakan fitur-fitur canggih yang ditawarkan dengan GameLift solusi Amazon yang dikelola sepenuhnya, yang juga menggunakan FleetIQ untuk meminimalkan biaya hosting. Jika Anda memerlukan fitur seperti perjodohan, perutean pemain berbasis latensi, sesi permainan dan manajemen sesi pemain, dan versi, lihat solusi Amazon. GameLift

Berikut adalah beberapa praktik terbaik yang dapat membantu Anda mendapatkan manfaat maksimal dari Amazon GameLift FleetIQ.

- Gunakan Amazon GameLift FleetIQ untuk game berbasis sesi. Amazon GameLift FleetIQ bekerja paling baik ketika terus-menerus mengarahkan pemain ke instans yang paling tidak mungkin mengalami interupsi sesi game. Mempertahankan sesi berumur panjang mengganggu proses penyeimbangan Amazon GameLift FleetIQ, yang meningkatkan kemungkinan sesi game mungkin terganggu. Alur kerja yang ideal adalah bagi pemain untuk beralih dari matchmaking (atau pemilihan server) ke dalam gameplay. Ketika game berakhir, pemain kembali ke matchmaking dan diarahkan ke server game lain pada instans baru. Kami merekomendasikan penggunaan Amazon GameLift FleetIQ untuk game dengan sesi di bawah dua jam.
- Berikan banyak jenis instans untuk dipilih. Ketika Anda menyiapkan grup server game, Anda memberikan daftar jenis instans yang akan digunakan. Semakin banyak jenis instans yang Anda sertakan, semakin besar fleksibilitas yang dimiliki Amazon GameLift FleetIQ untuk menggunakan Instans Spot dengan viabilitas tinggi untuk hosting game. Misalnya, Anda mungkin mencantumkan beberapa ukuran dalam keluarga instans yang sama (c5.large, c5.xlarge, c5.2xlarge, c5.4xlarge). Dengan instans yang lebih besar, Anda dapat menjalankan lebih banyak server game pada setiap instans, yang berpotensi menurunkan biaya. Dengan instans yang lebih kecil, penskalaan otomatis dapat bereaksi lebih cepat terhadap perubahan permintaan pemain. Perlu diingat bahwa daftar jenis instans yang diinginkan tidak diprioritaskan—grup Auto Scaling akan menggunakan keseimbangan jenis instans yang layak untuk menjaga ketahanan grup.
- Uji permainan Anda pada semua jenis instans. Pastikan bahwa server game Anda berjalan dengan baik pada setiap jenis instans yang Anda konfigurasi untuk grup server game Anda.
- Gunakan pembobotan kapasitas instans. Jika Anda mengonfigurasi grup server game untuk menggunakan berbagai ukuran instans (seperti c5.2xlarge, c5.4xlarge, c5.12xlarge), sertakan informasi pembobotan kapasitas untuk setiap jenis instans. Untuk informasi selengkapnya, lihat [Pembobotan Instans untuk Amazon EC2 Auto Scaling](#) dalam Panduan Pengguna Amazon EC2 Auto Scaling.
- Tempatkan sesi game Anda menggunakan Amazon GameLift FleetIQ. Saat menempatkan grup pemain dengan server game, gunakan Amazon GameLift `APIClaimGameServer()`. Amazon GameLift FleetIQ menghindari menempatkan pemain pada instans dengan peluang interupsi sesi game yang lebih tinggi.
- Laporkan status server game ke Amazon GameLift FleetIQ. Laporkan status kesehatan dan pemanfaatan server secara berkala dengan Amazon GameLift `APIUpdateGameServer()`. Mempertahankan status server game yang akurat membantu Amazon GameLift FleetIQ

menempatkan gameplay dengan lebih efisien. Hal ini juga membantu menghindari penghentian instans dengan gameplay aktif selama aktivitas penyeimbangan Spot.

- Siapkan kebijakan penskalaan otomatis. Anda dapat membuat kebijakan penskalaan pelacakan target yang menjaga kapasitas hosting Anda berdasarkan pemanfaatan pemain dan permintaan yang diantisipasi. Metrik Amazon GameLift FleetIQ `PercentUtilizedGameServers` adalah ukuran berapa banyak kapasitas hosting Anda saat ini digunakan. Kebanyakan game ingin mempertahankan buffer server game yang tidak terpakai sehingga pemain baru bisa masuk ke game dengan cepat. Anda dapat membuat kebijakan penskalaan yang mempertahankan ukuran buffer tertentu, menambahkan atau menghapus instans sebagai pemain permintaan berfluktuasi. Untuk informasi selengkapnya, lihat [Kebijakan Penskalaan Pelacakan Target](#) dalam Panduan Pengguna Amazon EC2 Auto Scaling.
- Gunakan AWS akun yang berbeda untuk lingkungan pengembangan dan produksi. Memisahkan konfigurasi pengembangan dan produksi Anda di akun yang berbeda dapat mengurangi risiko kesalahan konfigurasi yang berdampak pada pemain live.
- Aktifkan perlindungan sesi game untuk grup server game dalam produksi. Untuk melindungi pemain Anda, aktifkan perlindungan sesi game dan hindari sesi game aktif dihentikan lebih awal karena aktivitas penskalaan atau penyeimbangan.
- Uji game Anda di EC2 sebelum mengintegrasikannya dengan Amazon GameLift FleetIQ. Kami merekomendasikan agar game Anda berjalan di EC2 dan menyempurnakan konfigurasi Anda terlebih dahulu. Anda kemudian dapat membuat grup server game menggunakan templat peluncuran dan AMI yang sama.

Jika Anda menggunakan Kubernetes, kami rekomendasikan Anda terlebih dahulu mendapatkan instans EC2 standar yang ditambahkan ke klaster Kubernetes Anda, lalu buat grup server game menggunakan templat peluncuran yang Anda buat untuk node pekerja di klaster Kubernetes Anda. Jika Anda menggunakan EKS, buat klaster EKS dan grup server game secara terpisah. Untuk grup server game, gunakan AMI yang dioptimalkan EKS dengan data pengguna yang sesuai dan konfigurasi templat peluncuran yang digunakan untuk integrasi EKS Anda. Lihat detail lebih lanjut tentang simpul pekerja EKS dan AMI yang dioptimalkan EKS dalam panduan [AMI Linux yang dioptimalkan Amazon EKS](#).

- Gunakan strategi penyeimbangan grup server game **ON\_DEMAND\_ONLY** untuk ketersediaan server game yang andal. Dengan strategi penyeimbangan ini, tidak ada Instans Spot yang digunakan. Ini adalah alat yang berguna untuk memastikan ketersediaan server ketika Anda membutuhkannya, seperti selama peluncuran fitur atau kejadian khusus lainnya. Anda dapat berpindah grup server game dari Spot ke strategi Sesuai Permintaan sesuai keperluan.

Tinjau juga praktik terbaik AWS:

- [Praktik Terbaik untuk Amazon EC2](#)
- [Praktik Terbaik untuk Amazon EC2 Auto Scaling](#)

## Fitur Amazon GameLift FleetIQ

- **Penyeimbangan Spot yang Dioptimalkan.** Amazon GameLift FleetIQ secara berkala mengevaluasi jenis instans Anda dan menggantikan Instans Spot yang dianggap tidak layak karena potensi interupsi sesi game yang lebih tinggi. Saat grup EC2 Auto Scaling Anda menghentikan instans lama dan memulai yang baru, grup ini terus menyegarkan dengan jenis instans yang saat ini layak untuk hosting game.
- **Perutean pemain yang optimal.** Amazon GameLift FleetIQ API mengarahkan sesi game baru ke Instans Spot yang paling tangguh, yang paling tidak mungkin terganggu. Selain itu, sesi game dikemas ke instance yang lebih sedikit, yang meningkatkan kemampuan grup EC2 Auto Scaling untuk menurunkan sumber daya yang tidak dibutuhkan dan menurunkan biaya hosting.
- **Penskalaan otomatis berdasarkan penggunaan pemain.** Amazon GameLift FleetIQ memancarkan data pemanfaatan server game sebagai metrik Amazon. CloudWatch Anda dapat menggunakan metrik ini untuk secara otomatis menskalakan sumber daya hosting Anda yang tersedia untuk melacak dengan permintaan pemain yang sebenarnya dan mengurangi biaya hosting.
- **Manajemen langsung instans Amazon EC2.** Pertahankan kontrol penuh atas instans EC2 dan grup Penskalaan Otomatis EC2 di grup Anda. Akun AWS Ini berarti Anda dapat menyiapkan templat peluncuran instans, mempertahankan konfigurasi grup Penskalaan Otomatis EC2, dan berintegrasi dengan layanan lain. AWS Sebagai bagian dari aktivitas penyeimbangan Spot, Amazon GameLift FleetIQ membuat pembaruan berkala ke beberapa properti grup Penskalaan Otomatis EC2. Anda dapat mengganti pengaturan ini untuk sementara atau menanggukkan aktivitas Amazon GameLift FleetIQ sesuai kebutuhan.
- **Dukungan untuk beberapa format executable server game.** Amazon GameLift FleetIQ mendukung semua format yang saat ini berjalan di Amazon EC2, termasuk Windows, Linux, kontainer, dan Kubernetes. Lihat [FAQ Amazon EC2](#) untuk daftar sistem operasi dan waktu aktif yang didukung.
- **Beberapa jenis sumber daya hosting.** Dengan Amazon GameLift FleetIQ, Anda memiliki akses ke berbagai jenis instans untuk hosting server game. (Ketersediaan bervariasi menurut AWS Wilayah.) Ini berarti bahwa Anda dapat memasang server game Anda dengan campuran CPU, memori, penyimpanan, dan kapasitas jaringan yang sesuai untuk memberikan pengalaman gaming terbaik bagi pemain Anda.



- Jangkauan di seluruh dunia. Amazon GameLift FleetIQ tersedia di 15 Wilayah, termasuk di Tiongkok. Dengan jangkauan ini, Anda dapat membuat server game Anda tersedia dengan lag minimal untuk pemain, di mana pun mereka berada. Untuk daftar lengkap Wilayah, lihat [GameLift titik akhir dan kuota Amazon](#) di bagian. Referensi Umum AWS

## Harga untuk Amazon GameLift FleetIQ

Amazon GameLift mengenakan biaya untuk instans berdasarkan durasi penggunaan dan bandwidth menurut jumlah data yang ditransfer. Untuk daftar lengkap biaya dan harga Amazon GameLift, lihat [GameLift Harga Amazon](#).

Untuk informasi tentang menghitung biaya hosting game atau perbandingan Anda dengan Amazon GameLift, lihat [Menghasilkan perkiraan GameLift harga Amazon](#), yang menjelaskan cara menggunakan [AWS Pricing Calculator](#)

# Pengaturan Amazon GameLift FleetIQ

Topik di bagian ini membantu mengatur tugas, termasuk cara menyiapkan AWS akun Anda untuk digunakan dengan layanan Amazon Amazon GameLift FleetIQ.

Topik

- [Perangkat lunak yang GameLift didukung Amazon FleetIQ](#)
- [Siapkan AWS akun Anda untuk Amazon GameLift FleetIQ](#)

## Perangkat lunak yang GameLift didukung Amazon FleetIQ

Amazon GameLift FleetIQ digunakan untuk menerapkan server game multipemain, klien, dan layanan game 64-bit untuk hosting di Amazon EC2. Solusi ini mendukung lingkungan berikut:

Sistem operasi untuk server game

Anda dapat menggunakan Amazon GameLift FleetIQ dengan server game yang berjalan di salah satu sistem operasi yang didukung oleh EC2. Ini termasuk Amazon Linux, Ubuntu, Windows Server, Red Hat Enterprise Linux, SUSE Linux Enterprise Server, Fedora, Debian, CentOS, Gentoo Linux, Oracle Linux, dan FreeBSD. Lihat fitur dan dukungan EC2 saat ini di [Fitur Amazon EC2](#).

Penggunaan kontainer

Jika server game Anda menggunakan kontainer, Amazon GameLift FleetIQ mendukung integrasi dengan Kubernetes, Amazon Elastic Container Service (Amazon ECS), dan Amazon Elastic Kubernetes Service (EKS). Lihat informasi lebih lanjut di [Kontainer di AWS](#).

Lingkungan pengembangan game

Klien dan server game memerlukan beberapa integrasi untuk berkomunikasi dengan layanan Amazon GameLift FleetIQ. Game membuat panggilan API ke SDK AWS. [Unduh AWS SDK](#) atau [lihat dokumentasi referensi Amazon GameLift API](#).

AWSSDK dengan dukungan untuk Amazon GameLift tersedia dalam bahasa berikut. Untuk informasi tentang dukungan untuk lingkungan pengembangan, lihat dokumentasi untuk setiap bahasa.

- [C ++ \(dokumen SDK\) \(Amazon\) GameLift](#)
- [Java \(dokumen SDK\) \(Amazon\) GameLift](#)
- [.NET \(dokumen SDK\) \(Amazon\) GameLift](#)

- [Pergi \(dokumen SDK\) \(Amazon\) GameLift](#)
- [Python \(dokumen SDK\) \(Amazon\) GameLift](#)
- [Ruby \(dokumen SDK\) \(Amazon\) GameLift](#)
- [PHP \(dokumen SDK\) \(Amazon\) GameLift](#)
- [JavaScript/Node.js \(dokumen SDK\) \(Amazon\) GameLift](#)

## Siapkan AWS akun Anda untuk Amazon GameLift FleetIQ

Untuk menggunakan Amazon GameLift FleetIQ dengan Amazon EC2, Auto Scaling, AWS dan layanan lainnya, Anda harus menyiapkan izin akses yang diperlukan. Akun AWS Lakukan hal-hal berikut:

- Jika Anda belum memiliki AWS akun untuk digunakan dengan Amazon GameLift FleetIQ, buat yang baru. Lihat [Buat sebuah Akun AWS](#).
- Tetapkan izin GameLift khusus Amazon FleetIQ untuk pengguna dan grup pengguna. Lihat [Mengelola izin pengguna untuk Amazon GameLift FleetIQ](#).
- Buat peran IAM untuk memungkinkan Amazon GameLift dan sumber daya Amazon EC2 Anda berinteraksi. Lihat [Membuat IAM role untuk interaksi lintas layanan](#).

## Buat sebuah Akun AWS

Buat dan atur Akun AWS untuk digunakan dengan Amazon GameLift FleetIQ. Tidak ada biaya untuk membuat Akun AWS.

Topik

- [Mendaftar untuk Akun AWS](#)
- [Buat pengguna dengan akses administratif](#)

## Mendaftar untuk Akun AWS

Jika Anda tidak memiliki Akun AWS, selesaikan langkah-langkah berikut untuk membuatnya.

Untuk mendaftar untuk Akun AWS

1. Buka <https://portal.aws.amazon.com/billing/signup>.

## 2. Ikuti petunjuk online.

Bagian dari prosedur pendaftaran melibatkan tindakan menerima panggilan telepon dan memasukkan kode verifikasi di keypad telepon.

Saat Anda mendaftar untuk sebuah Akun AWS, sebuah Pengguna root akun AWS dibuat. Pengguna root memiliki akses ke semua Layanan AWS dan sumber daya di akun. Sebagai praktik keamanan terbaik, tetapkan akses administratif ke pengguna, dan gunakan hanya pengguna root untuk melakukan [tugas yang memerlukan akses pengguna root](#).

AWS mengirimkan Anda email konfirmasi setelah proses pendaftaran selesai. Anda dapat melihat aktivitas akun Anda saat ini dan mengelola akun Anda dengan mengunjungi <https://aws.amazon.com/> dan memilih Akun Saya.

## Buat pengguna dengan akses administratif

Setelah Anda mendaftar Akun AWS, amankan Pengguna root akun AWS, aktifkan AWS IAM Identity Center, dan buat pengguna administratif sehingga Anda tidak menggunakan pengguna root untuk tugas sehari-hari.

### Amankan Anda Pengguna root akun AWS

1. Masuk ke [AWS Management Console](#) sebagai pemilik akun dengan memilih pengguna Root dan memasukkan alamat Akun AWS email Anda. Di laman berikutnya, masukkan kata sandi.

Untuk bantuan masuk dengan menggunakan pengguna root, lihat [Masuk sebagai pengguna root](#) di AWS Sign-In Panduan Pengguna.

2. Mengaktifkan autentikasi multi-faktor (MFA) untuk pengguna root Anda.

Untuk petunjuk, lihat [Mengaktifkan perangkat MFA virtual untuk pengguna Akun AWS root \(konsol\) Anda](#) di Panduan Pengguna IAM.

### Buat pengguna dengan akses administratif

1. Aktifkan Pusat Identitas IAM.

Untuk mendapatkan petunjuk, silakan lihat [Mengaktifkan AWS IAM Identity Center](#) di Panduan Pengguna AWS IAM Identity Center .

2. Di Pusat Identitas IAM, berikan akses administratif ke pengguna.

Untuk tutorial tentang menggunakan Direktori Pusat Identitas IAM sebagai sumber identitas Anda, lihat [Mengkonfigurasi akses pengguna dengan default Direktori Pusat Identitas IAM](#) di Panduan AWS IAM Identity Center Pengguna.

Masuk sebagai pengguna dengan akses administratif

- Untuk masuk dengan pengguna Pusat Identitas IAM, gunakan URL masuk yang dikirim ke alamat email saat Anda membuat pengguna Pusat Identitas IAM.

Untuk bantuan masuk menggunakan pengguna Pusat Identitas IAM, lihat [Masuk ke portal AWS akses](#) di Panduan AWS Sign-In Pengguna.

Tetapkan akses ke pengguna tambahan

1. Di Pusat Identitas IAM, buat set izin yang mengikuti praktik terbaik menerapkan izin hak istimewa paling sedikit.

Untuk petunjuknya, lihat [Membuat set izin](#) di Panduan AWS IAM Identity Center Pengguna.

2. Tetapkan pengguna ke grup, lalu tetapkan akses masuk tunggal ke grup.

Untuk petunjuk, lihat [Menambahkan grup](#) di Panduan AWS IAM Identity Center Pengguna.

## Mengelola izin pengguna untuk Amazon GameLift FleetIQ

Buat pengguna tambahan atau perluas izin akses Amazon GameLift FleetIQ ke pengguna yang ada sesuai kebutuhan. Pengguna yang bekerja dengan grup server game Amazon GameLift FleetIQ dan layanan Amazon EC2 dan Auto Scaling terkait harus memiliki izin untuk mengakses layanan ini.

Sebagai praktik terbaik (Praktik [terbaik keamanan di IAM](#)), terapkan izin hak istimewa paling sedikit untuk semua pengguna. Anda dapat mengatur izin untuk pengguna individu atau grup pengguna dan membatasi akses pengguna berdasarkan layanan, tindakan, atau sumber daya.

Gunakan petunjuk berikut untuk menetapkan izin pengguna berdasarkan cara Anda mengelola pengguna di AWS akun Anda. Jika Anda menggunakan pengguna IAM, sebagai praktik terbaik selalu lampirkan izin ke peran atau grup pengguna, bukan pengguna individu.

- [Sintaks izin untuk pengguna](#)

- [Sintaks izin tambahan untuk digunakan dengan AWS CloudFormation](#)

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

- Pengguna dan grup di AWS IAM Identity Center:

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

- Pengguna yang dikelola di IAM melalui penyedia identitas:

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

- Pengguna IAM:

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

## Amazon GameLift FleetIQ\_Policy

```
{
  "Version": "2012-10-17",
  "Statement":
  [
    {
      "Action":
      [
        "iam:PassRole"
      ],
      "Effect": "Allow",
      "Resource": "*",
      "Condition":
      {
        "StringEquals":
        {
          "iam:PassedToService": "gamelift.amazonaws.com"
        }
      }
    }
  ]
}
```

```
    },
    {
      "Action":
      [
        "iam:CreateServiceLinkedRole"
      ],
      "Effect": "Allow",
      "Resource": "arn:*:iam:*:*:role/aws-service-role/autoscaling.amazonaws.com/
AWSServiceRoleForAutoScaling"
    },
    {
      "Action":
      [
        "autoscaling:CreateAutoScalingGroup",
        "autoscaling:CreateOrUpdateTags",
        "autoscaling:DescribeAutoScalingGroups",
        "autoscaling:ExitStandby",
        "autoscaling:PutLifecycleHook",
        "autoscaling:PutScalingPolicy",
        "autoscaling:ResumeProcesses",
        "autoscaling:SetInstanceProtection",
        "autoscaling:UpdateAutoScalingGroup",
        "autoscaling>DeleteAutoScalingGroup"
      ],
      "Effect": "Allow",
      "Resource": "*"
    },
    {
      "Action":
      [
        "ec2:DescribeAvailabilityZones",
        "ec2:DescribeSubnets",
        "ec2:RunInstances",
        "ec2:CreateTags"
      ],
      "Effect": "Allow",
      "Resource": "*"
    },
    {
      "Action":
      [
        "events:PutRule",
        "events:PutTargets"
      ],
    },
```

```

    "Effect": "Allow",
    "Resource": "*"
  }
]
}

```

## Izin tambahan untuk AWS CloudFormation

Jika Anda menggunakannya AWS CloudFormation untuk mengelola sumber daya hosting game, tambahkan AWS CloudFormation izin ke sintaks kebijakan.

```

{
  "Action": [
    "autoscaling:DescribeLifecycleHooks",
    "autoscaling:DescribeNotificationConfigurations",
    "ec2:DescribeLaunchTemplateVersions"
  ]
  "Effect": "Allow",
  "Resource": "*"
}

```

## Menyiapkan akses terprogram untuk pengguna

Pengguna membutuhkan akses terprogram jika mereka ingin berinteraksi dengan AWS luar. AWS Management Console Cara untuk memberikan akses terprogram tergantung pada jenis pengguna yang mengakses AWS.

Untuk memberi pengguna akses programatis, pilih salah satu opsi berikut.

Pengguna mana yang membutuhkan akses programatis?	Untuk	Oleh
Identitas tenaga kerja (Pengguna yang dikelola di Pusat Identitas IAM)	Gunakan kredensial sementara untuk menandatangani permintaan terprogram ke AWS CLI, AWS SDK, atau API. AWS	Mengikuti petunjuk untuk antarmuka yang ingin Anda gunakan. <ul style="list-style-type: none"> <li>Untuk AWS CLI, lihat <a href="#">Mengkonfigurasi yang akan AWS CLI digunakan AWS</a></li> </ul>



Pengguna mana yang membutuhkan akses programatis?	Untuk	Oleh
		<p><a href="#">IAM Identity Center</a> dalam Panduan AWS Command Line Interface Pengguna.</p> <ul style="list-style-type: none"><li>• Untuk AWS SDK, alat, dan AWS API, lihat <a href="#">otentikasi Pusat Identitas IAM</a> di Panduan Referensi AWS SDK dan Alat.</li></ul>
IAM	Gunakan kredensial sementara untuk menandatangani permintaan terprogram ke AWS CLI, AWS SDK, atau API. AWS	Mengikuti petunjuk dalam <a href="#">Menggunakan kredensial sementara dengan AWS sumber daya</a> di Panduan Pengguna IAM.

Pengguna mana yang membutuhkan akses programatis?	Untuk	Oleh
IAM	(Tidak direkomendasikan) Gunakan kredensial jangka panjang untuk menandatangani permintaan terprogram ke AWS CLI, AWS SDK, atau API. AWS	<p>Mengikuti petunjuk untuk antarmuka yang ingin Anda gunakan.</p> <ul style="list-style-type: none"> <li>• Untuk mengetahui AWS CLI, lihat <a href="#">Mengautentikasi menggunakan kredensial pengguna IAM di Panduan Pengguna</a>. AWS Command Line Interface</li> <li>• Untuk AWS SDK dan alat bantu, lihat <a href="#">Mengautentikasi menggunakan kredensial jangka panjang di Panduan Referensi AWS</a> SDK dan Alat.</li> <li>• Untuk AWS API, lihat <a href="#">Mengelola kunci akses untuk pengguna IAM di Panduan Pengguna IAM</a>.</li> </ul>

Jika Anda menggunakan kunci akses, lihat [Praktik terbaik untuk mengelola kunci AWS akses](#).

## Membuat IAM role untuk interaksi lintas layanan

Agar Amazon GameLift FleetIQ dapat bekerja dengan instans Amazon EC2 dan grup Auto Scaling, Anda harus mengizinkan layanan berinteraksi satu sama lain. Hal ini dilakukan dengan membuat IAM role di akun AWS Anda dan menetapkan satu set izin terbatas. Setiap peran juga menentukan layanan mana yang dapat mengambil peran tersebut.

Siapkan peran berikut ini:

- [Buat peran untuk Amazon GameLift FleetIQ](#) untuk memperbarui sumber daya Amazon EC2 Anda.

- [Buat peran untuk Amazon EC2](#) sumber daya untuk berkomunikasi dengan Amazon GameLift FleetIQ.

## Buat peran untuk Amazon GameLift FleetIQ

Peran ini memungkinkan Amazon GameLift FleetIQ mengakses dan memodifikasi instans Amazon EC2, grup Auto Scaling, dan kait siklus hidup Anda sebagai bagian dari aktivitas penyeimbangan Spot dan penskalaan otomatisnya.

Gunakan konsol IAM atau AWS CLI untuk membuat peran Amazon GameLift FleetIQ dan lampirkan kebijakan terkelola dengan izin yang diperlukan. Untuk informasi selengkapnya tentang peran IAM dan kebijakan terkelola, lihat [Membuat Peran untuk AWS Layanan](#) dan [Kebijakan AWS Terkelola](#).

### Console

Langkah-langkah ini menjelaskan cara membuat peran layanan dengan kebijakan terkelola untuk Amazon GameLift menggunakan AWS Management Console.

1. Buka [Konsol IAM](#) dan pilih Peran: Buat peran.
2. Untuk Pilih tipe entitas tepercaya, pilih Layanan AWS .
3. Untuk Pilih kasus penggunaan, pilih GameLift dari daftar layanan. Di bawah Pilih kasus penggunaan Anda, kasus GameLift penggunaan Amazon yang sesuai dipilih secara otomatis. Untuk melanjutkan, pilih Berikutnya: Izin.
4. Daftar Kebijakan izin terlampir harus berisi satu kebijakan: GameLiftGameServerGroupPolicy . Jika kebijakan ini tidak ditampilkan, periksa filter atau gunakan fitur pencarian untuk menambahkannya ke peran. Anda dapat melihat sintaks kebijakan (pilih ikon ► untuk memperluas), tetapi Anda tidak dapat mengubah sintaks. Saat peran dibuat, Anda dapat memperbarui peran dan melampirkan kebijakan tambahan untuk menambah atau menghapus izin.

Untuk Atur batas izin, gunakan pengaturan default (Buat peran tanpa batas izin). Ini adalah pengaturan lanjutan yang tidak diperlukan. Untuk melanjutkan, pilih Selanjutnya: Tanda.

5. Tambahkan tanda adalah pengaturan opsional untuk manajemen sumber daya. Misalnya, Anda mungkin ingin menambahkan tanda ke peran ini untuk melacak penggunaan sumber daya khusus proyek berdasarkan peran. Untuk melihat informasi lebih lanjut tentang penandaan untuk IAM role dan penggunaan lainnya, klik tautan Pelajari selengkapnya. Untuk melanjutkan, pilih Selanjutnya: Tinjauan.

6. Pada halaman Tinjauan, lakukan perubahan berikut sesuai kebutuhan:
  - Masukkan nama peran dan secara opsional perbarui deskripsi.
  - Verifikasi hal berikut:
    - Entitas tepercaya diatur ke "AWS layanan: gamelift.amazonaws.com". Nilai ini harus diperbarui setelah peran dibuat.
    - Kebijakan termasuk GameLiftGameServerGroupPolicy.

Untuk menyelesaikan tugas, pilih Buat peran.

7. Setelah peran baru dibuat, Anda harus memperbarui relasi kepercayaan peran secara manual. Buka Peran dan pilih nama peran baru untuk membuka halaman ringkasannya. Pilih tab Relasi kepercayaan, dan kemudian pilih Edit relasi kepercayaan. Dalam dokumen kebijakan, perbarui properti Service untuk menyertakan `autoscaling.amazonaws.com`. Properti Service yang telah direvisi akan terlihat seperti ini:

```
"Service": [  
  "gamelift.amazonaws.com",  
  "autoscaling.amazonaws.com"  
]
```

Untuk menyimpan perubahan, pilih Perbarui Trust Policy.

Peran ini sekarang siap digunakan. Perhatikan nilai ARN peran, yang ditampilkan di bagian atas halaman ringkasan peran. Anda akan memerlukan informasi ini saat mengatur grup server game Amazon GameLift FleetIQ.

## AWS CLI

Langkah-langkah ini menjelaskan cara membuat peran layanan dengan kebijakan terkelola untuk Amazon GameLift menggunakan AWS CLI.

1. Membuat file trust policy (contoh: `FleetIQtrustpolicyGameLift.json`) dengan sintaks JSON berikut.

```
{  
  "Version": "2012-10-17",  
  "Statement": [  
    {
```

```
"Effect": "Allow",
"Principal": {
  "Service": [
    "gamelift.amazonaws.com",
    "autoscaling.amazonaws.com"
  ]
},
"Action": "sts:AssumeRole"
}
]
}
```

2. Buat IAM role baru dengan [iam create-role](#) dan kaitkan dengan file JSON trust policy yang baru saja Anda buat.

Windows:

```
AWS iam create-role --role-name FleetIQ-role-for-GameLift --assume-role-policy-document file://C:\policies\FleetIQtrustpolicyGameLift.json
```

Linux:

```
AWS iam create-role --role-name FleetIQ-role-for-GameLift --assume-role-policy-document file://policies/FleetIQtrustpolicyGameLift.json
```

Ketika permintaan berhasil, respon meliputi properti dari peran yang baru dibuat. Perhatikan nilai ARN. Anda akan memerlukan informasi ini saat mengatur grup server game Amazon GameLift FleetIQ.

3. Gunakan [iam attach-role-policy](#) untuk melampirkan kebijakan izin terkelola `"GameLiftGameServerGroupPolicy"`.

```
AWS iam attach-role-policy --role-name FleetIQ-role-for-GameLift --policy-arn arn:aws:iam::aws:policy/GameLiftGameServerGroupPolicy
```

Untuk memverifikasi bahwa kebijakan izin dilampirkan, panggil [iam list-attached-role-policies](#) dengan nama peran baru.

Peran ini sekarang siap digunakan. Anda dapat memverifikasi bahwa peran IAM dikonfigurasi dengan benar dengan memanggil [gamelift create-game-server-group](#) dengan `role-arn` properti

yang disetel ke nilai ARN peran baru. Saat `GameServerGroup` memasuki status AKTIF, ini menunjukkan bahwa Amazon GameLift FleetIQ dapat memodifikasi sumber daya Amazon EC2 dan Auto Scaling di akun Anda, seperti yang diharapkan.

## Buat peran untuk Amazon EC2

Peran ini memungkinkan sumber daya Amazon EC2 Anda untuk berkomunikasi dengan Amazon GameLift FleetIQ. Misalnya, server game Anda, yang berjalan pada instans Amazon EC2, harus dapat melaporkan status kondisi. Sertakan peran ini dalam profil instans IAM dengan template peluncuran Amazon EC2 Anda saat membuat grup server game Amazon GameLift FleetIQ.

Gunakan AWS CLI untuk membuat peran untuk Amazon EC2, lampirkan kebijakan khusus dengan izin yang diperlukan, dan lampirkan peran tersebut ke profil instans. Untuk informasi selengkapnya, lihat [Membuat Peran untuk AWS Layanan](#).

### AWS CLI

Langkah-langkah ini menjelaskan cara membuat peran layanan dengan GameLift izin Amazon khusus untuk Amazon EC2 menggunakan AWS CLI

1. Membuat file trust policy (contoh: `FleetIQtrustpolicyEC2.json`) dengan sintaks JSON berikut.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "ec2.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

2. Buat IAM role baru dengan [iam create-role](#) dan kaitkan dengan file JSON trust policy yang baru saja Anda buat.

Windows:

```
AWS iam create-role --role-name FleetIQ-role-for-EC2 --assume-role-policy-document file://C:\policies\FleetIQtrustpolicyEC2.json
```

Linux:

```
AWS iam create-role --role-name FleetIQ-role-for-EC2 --assume-role-policy-document file://policies/FleetIQtrustpolicyEC2.json
```

Ketika permintaan berhasil, respon meliputi properti dari peran yang baru dibuat. Perhatikan nilai ARN. Anda akan membutuhkan informasi ini ketika menyiapkan templat peluncuran Amazon EC2 Anda.

3. Membuat file kebijakan izin (contoh: `FleetIQpermissionsEC2.json`) dengan sintaks JSON berikut.

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

4. Gunakan [iam put-role-policy](#) untuk melampirkan file JSON kebijakan izin, yang baru saja Anda buat, ke peran baru.

Windows:

```
AWS iam put-role-policy --role-name FleetIQ-role-for-EC2 --policy-name FleetIQ-permissions-for-EC2 --policy-document file://C:\policies\FleetIQpermissionsEC2.json
```

Linux:

```
AWS iam put-role-policy --role-name FleetIQ-role-for-EC2 --policy-name FleetIQ-permissions-for-EC2 --policy-document file://policies/FleetIQpermissionsEC2.json
```

Untuk memverifikasi bahwa kebijakan izin dilampirkan, panggil [iam list-role-policies](#) dengan nama peran baru.

5. Buat profil instans dengan [iam create-instance-profile](#) dengan peran baru untuk digunakan dengan Amazon EC2. Untuk informasi selengkapnya, lihat [Mengelola Profil Instans](#).

```
AWS iam create-instance-profile --instance-profile-name FleetIQ-role-for-EC2
```

Ketika permintaan berhasil, respon meliputi properti dari profil instans yang baru dibuat.

6. Gunakan [iam add-role-to-instance-profile](#) untuk melampirkan peran ke profil instance.

```
AWS iam add-role-to-instance-profile --role-name FleetIQ-role-for-EC2 --instance-profile-name FleetIQ-role-for-EC2
```

Peran dan profil sekarang siap untuk digunakan dengan templat peluncuran Amazon EC2.



# Mempersiapkan game untuk Amazon GameLift FleetIQ

Bagian ini mencakup cara menerapkan desain Anda untuk hosting game di Amazon EC2 dengan Amazon GameLift FleetIQ Untuk menjalankan dan menjalankan game multipemain Anda, Anda perlu melakukan hal berikut:

- Sesuaikan server game Anda untuk berkomunikasi dengan Amazon GameLift FleetIQ.
- Buat grup server game FleetIQ untuk men-deploy server game Anda.
- Tambahkan fungsionalitas ke layanan client game Anda untuk meminta server game yang tersedia.

Topik dalam bagian ini memberikan informasi detail tentang cara menyelesaikan pekerjaan ini. Untuk memulai, lihat rencana integrasi, yang menyediakan step-by-step panduan terperinci.

Topik

- [Langkah-langkah integrasi Amazon GameLift FleetIQ](#)
- [Mengelola grup GameLift server game Amazon FleetIQ](#)
- [Integrasikan Amazon GameLift FleetIQ ke dalam server game](#)
- [Integrasikan Amazon GameLift FleetIQ ke dalam klien game](#)

## Langkah-langkah integrasi Amazon GameLift FleetIQ

Rencana integrasi ini menguraikan langkah-langkah kunci untuk membuat game multipemain Anda aktif dan berjalan di instans Amazon EC2 dengan Amazon FleetIQ. GameLift Jika Anda mencari layanan hosting GameLift terkelola Amazon, yang mengotomatiskan lebih banyak proses hosting game untuk Anda, lihat [Panduan GameLift Pengembang Amazon](#).

Untuk mulai menggunakan Amazon GameLift FleetIQ, Anda harus memiliki server game yang berfungsi yang berjalan baik di lingkungan lokal atau Amazon EC2. Server game Anda bisa menjadi proses tunggal yang mengelola satu atau beberapa sesi game, memunculkan proses anak, atau berjalan di dalam kontainer.

1. Dapatkan [AWSakun](#) dan siapkan pengguna dengan akses Amazon GameLift FleetIQ.

Buat akun baru Akun AWS atau pilih akun yang ada untuk digunakan dengan Amazon GameLift FleetIQ. Siapkan pengguna dengan izin untuk mengelola Amazon EC2, Auto Scaling, dan AWS

sumber daya lain yang digunakan dengan game Anda. Untuk instruksi detail, lihat [Siapkan AWS akun Anda untuk Amazon GameLift FleetIQ](#).

## 2. Buat peran IAM.

Buat peran yang memungkinkan sumber daya Amazon GameLift FleetIQ, Amazon EC2, dan Auto Scaling untuk berkomunikasi satu sama lain. Untuk detail lebih lanjut, lihat [Membuat IAM role untuk interaksi lintas layanan](#).

## 3. Dapatkan AWS SDK dan AWS CLI dengan fungsionalitas Amazon GameLift FleetIQ.

- [Unduh versi terbaru dari SDK AWS](#).
- [Lihat dokumentasi referensi GameLift API Amazon](#).

## 4. Siapkan server game Anda untuk digunakan dengan Amazon GameLift FleetIQ.

Tambahkan AWS SDK ke proyek server game Anda dan tambahkan kode agar Amazon GameLift FleetIQ tetap diperbarui dengan status dan penggunaan server game Anda saat ini. Lihat [the section called “Mengintegrasikan server game”](#) untuk panduan dan contoh tambahan. Amazon GameLift FleetIQ menggunakan informasi ini untuk menyediakan sistem perjadwalan Anda dengan daftar server game yang layak dan tidak dihuni, dan juga menghindari penghentian instans yang saat ini menjadi tuan rumah pemain selama penyeimbangan.

## 5. Buat Amazon EC2 Amazon Machine Image (AMI) dengan server game Anda.

Buat AMI dengan perangkat lunak server game Anda dan dengan aset waktu aktif atau pengaturan konfigurasi lainnya. Untuk bantuan, lihat [Amazon Machine Images \(AMI\)](#) dalam Panduan Pengguna Amazon EC2.

## 6. Buat template peluncuran Amazon EC2.

Membangun templat peluncuran Amazon EC2 yang menggunakan AMI kustom Anda dan mendefinisikan jaringan dan pengaturan keamanan untuk sumber daya hosting Anda. Templat peluncuran harus mereferensikan profil instans yang Anda buat (lihat Langkah 2) dengan izin yang memungkinkan server game Anda berkomunikasi dengan Amazon GameLift FleetIQ. Anda tidak perlu menyertakan jenis instans dalam templat peluncuran Anda, karena hal ini dilakukan nanti. Untuk bantuan, lihat [Membuat Templat Peluncuran](#) dalam Panduan Pengguna Amazon EC2.

**Note**

Sebelum menggunakan template peluncuran dengan Amazon GameLift FleetIQ, kami sangat menyarankan Anda terlebih dahulu menyiapkan grup Penskalaan Otomatis untuk memverifikasi bahwa konfigurasi template dan AMI diterapkan dengan benar.

## 7. Siapkan sumber daya hosting Amazon GameLift FleetIQ.

Di setiap Wilayah tempat Anda ingin menerapkan server game, buat grup server game dengan memanggil [CreateGameServerGroup\(\)](#). Masukkan templat peluncuran (berisi pengaturan AMI dan jaringan dan keamanan kustom Anda), IAM role, dan daftar jenis instans yang dapat dijalankan game Anda. Tindakan ini menyiapkan grup Penskalaan Otomatis di AWS akun Anda yang dapat diubah oleh Amazon GameLift FleetIQ. Untuk panduan dan contoh tambahan, lihat [Mengelola grup GameLift server game Amazon FleetIQ](#).

## 8. Integrasikan Amazon GameLift FleetIQ ke klien game Anda.

Tambahkan SDK AWS ke client game, matchmaker, atau komponen backend lainnya yang mengalokasikan kapasitas server game. Tergantung pada jenis permainan Anda, mak comblang Anda mungkin menelepon [ListGameServers\(\)](#) atau [ClaimGameServer\(\)](#) untuk mendapatkan kapasitas server dan memesan server game yang tersedia. Untuk panduan dan contoh tambahan, lihat [Integrasikan Amazon GameLift FleetIQ ke dalam klien game](#).

## 9. Tingkatkan skala grup Auto Scaling Anda.

Karena instans disediakan dalam grup Auto Scaling Anda, instans meluncurkan server game Anda. Setiap server game kemudian mendaftar dengan Amazon GameLift FleetIQ sebagai kapasitas yang tersedia, untuk dicantumkan atau diklaim nanti oleh mak comblang Anda.

## 10. Uji permainan Anda.

Panggil matchmaker Anda dan panggil `ClaimGameServer` untuk meminta kapasitas server. Masukkan IP dan port yang dihasilkan kembali ke client game sehingga dapat terhubung ke server game.

# Mengelola grup GameLift server game Amazon FleetIQ

Topik ini menjelaskan tugas yang diperlukan untuk menyiapkan grup server game Amazon GameLift FleetIQ. Membuat grup server game memicu pembuatan grup Penskalaan Otomatis EC2 dengan

semua pengaturan konfigurasi yang diperlukan, bersama dengan konfigurasi untuk mengelola pengoptimalan Amazon GameLift FleetIQ untuk hosting game.

Sebelum Anda dapat membuat grup server game, Anda harus setidaknya menyiapkan sumber daya berikut:

- Templat peluncuran Amazon EC2 yang menentukan bagaimana memulai instans Amazon EC2 dengan build server game Anda. Untuk informasi selengkapnya, lihat [Meluncurkan Instans dari Templat Peluncuran](#) dalam Panduan Pengguna Amazon EC2.
- Peran IAM yang memperluas akses terbatas ke AWS akun Anda untuk memungkinkan GameLift FleetIQ membuat dan berinteraksi dengan grup Auto Scaling. Untuk informasi selengkapnya, lihat [Membuat IAM role untuk interaksi lintas layanan](#).

## Membuat grup server game

Untuk membuat grup server game, panggil [CreateGameServerGroup\(\)](#). Operasi ini membuat grup server game Amazon GameLift FleetIQ dan grup Penskalaan Otomatis yang sesuai. Saat Anda membuat grup server game, Anda memberikan pengaturan khusus game untuk Amazon GameLift FleetIQ, termasuk strategi penyeimbangan dan definisi jenis instans. Anda juga menyediakan pengaturan properti awal untuk grup Auto Scaling.

Instans berikut memicu pembuatan `GameServerGroup` yang menentukan jenis instans `c4.large` dan `c5.large` dan membatasi grup hanya untuk Instans Spot, dan grup Auto Scaling yang menggunakan templat peluncuran yang ditentukan untuk men-deploy instans dan mengelola kapasitas grup dalam pengaturan minimum dan maksimum menggunakan kebijakan penskalaan otomatis pelacakan target. Setelah periode penyediaan singkat, sebuah sumber daya `AutoScalingGroup` dibuat, dan `GameServerGroup` memasuki keadaan AKTIF.

```
AWS gamelift create-game-server-group \  
  --game-server-group-name MyLiveGroup \  
  --role-arn arn:aws:iam::123456789012:role/GameLiftGSGRole \  
  --min-size 1 \  
  --max-size 10 \  
  --game-server-protection-policy FULL_PROTECTION \  
  --balancing-strategy SPOT_ONLY \  
  --launch-template LaunchTemplateId=lt-012ab345cde6789ff \  
  --instance-definitions '[{"InstanceType": "c4.large"}, {"InstanceType":  
"c5.large"}]' \  
  --auto-scaling-policy '{"TargetTrackingConfiguration": {"TargetValue": 66}}'
```

## Memperbarui grup server game

Anda dapat memperbarui properti grup server game yang memengaruhi cara Amazon GameLift FleetIQ mengelola hosting untuk server game, termasuk pengoptimalan jenis sumber daya. Untuk memperbarui properti ini, panggil [UpdateGameServerGroup\(\)](#). Setelah perubahan pada grup server game berlaku, Amazon GameLift FleetIQ dapat menimpa properti tertentu dalam grup Penskalaan Otomatis.

Untuk semua properti grup Auto Scaling lainnya, seperti `MinSize`, `MaxSize`, dan `LaunchTemplate`, Anda dapat memodifikasinya langsung di grup Auto Scaling.

Pada contoh di bawah ini, definisi jenis instans diperbarui untuk beralih ke jenis instans `c4.xlarge` dan `c5.xlarge`.

```
AWS gamelift update-game-server-group \  
  --game-server-group-name MyLiveGroup \  
  --instance-definitions '[{"InstanceType": "c4.xlarge"}, {"InstanceType":  
  "c5.xlarge"}]'
```

## Melacak instans grup server game

[Setelah membuat dan menyebarkan instance ke grup server game dan grup Auto Scaling, Anda dapat melacak status instance server game dengan memanggil `\(\)`. `DescribeGameServerInstances`](#) Anda dapat menggunakan operasi ini untuk melacak status instance.. Untuk informasi selengkapnya tentang status grup server game, lihat [Kehidupan grup server game](#).

Anda juga dapat menggunakan [GameLiftkonsol Amazon](#), di bawah grup server Game, untuk memantau status grup server game Anda.

## Integrasikan Amazon GameLift FleetIQ ke dalam server game

Topik ini menjelaskan tugas yang diperlukan untuk mempersiapkan proyek server game Anda untuk berkomunikasi dengan Amazon GameLift FleetIQ. Lihat [Praktik terbaik Amazon GameLift FleetIQ](#) untuk petunjuk tambahan.

## Mendaftarkan server game

Ketika proses server game diluncurkan dan siap untuk menjadi tuan rumah gameplay langsung, ia harus mendaftar dengan Amazon GameLift FleetIQ dengan menelepon [RegisterGameServer\(\)](#).

Pendaftaran memungkinkan Amazon GameLift FleetIQ merespons sistem perجدohan atau layanan klien lainnya ketika mereka meminta informasi tentang kapasitas server atau mengklaim server game. Saat mendaftar, server game dapat menyediakan data server game dan informasi koneksi yang relevan kepada Amazon GameLift FleetIQ, termasuk port dan alamat IP yang digunakannya untuk koneksi klien masuk.

```
AWS gamelift register-game-server \  
  --game-server-id UniqueId-1234 \  
  --game-server-group-name MyLiveGroup \  
  --instance-id i-1234567890 \  
  --connection-info "1.2.3.4:123" \  
  --game-server-data "{\"key\": \"value\"}"
```

## Memperbarui status server game

Setelah server game terdaftar, server harus secara teratur melaporkan status kesehatan dan pemanfaatan untuk menjaga status kapasitas server tetap sinkron di Amazon GameLift FleetIQ. Laporkan status kesehatan dan pemanfaatan dengan menelepon [UpdateGameServer\(\)](#). Pada contoh di bawah ini, server game melaporkan bahwa kondisi baik dan saat ini tidak ditempati oleh hosting pemain atau gameplay.

```
AWS gamelift update-game-server \  
  --game-server-group-name MyLiveGroup \  
  --game-server-id UniqueId-1234 \  
  --health-check HEALTHY \  
  --utilization-status AVAILABLE
```

### Status kesehatan

Jika server game Anda memiliki mekanisme untuk melacak status kesehatan, Anda dapat menggunakan mekanisme ini untuk memicu pembaruan kesehatan server game ke Amazon GameLift FleetIQ.

### Status pemanfaatan

Pelaporan status pemanfaatan server game membuat Amazon GameLift FleetIQ mendapatkan informasi tentang server game mana yang saat ini ideal dan tersedia untuk sesi game baru. Server game Anda harus memiliki mekanisme yang memicu pembaruan status pemanfaatan ke Amazon FleetIQ. GameLift Misalnya, Anda mungkin memicu pembaruan saat pemain tersambung ke server game atau saat sesi game dimulai.

Saat memulai sesi permainan, klien atau layanan perjudohan mengklaim server game yang tersedia (dengan menelepon [ClaimGameServer\(\)](#)), meminta pemain untuk terhubung ke server game, dan memicu server game untuk memulai permainan. Proses ini dijelaskan di [Integrasikan Amazon GameLift FleetIQ ke dalam klien game](#). Sebuah "klaim" server game berlaku selama 60 detik, dan server game harus dapat memperbarui status pemanfaatan selama rentang waktu ini. Jika status pemanfaatan tidak diperbarui, Amazon GameLift FleetIQ menghapus klaim, mengasumsikan bahwa server game tersedia, dan dapat memesan server game untuk permintaan klaim klien lain.

```
AWS gamelift update-game-server \  
  --game-server-group-name MyLiveGroup \  
  --game-server-id UniqueId-1234 \  
  --health-check HEALTHY \  
  --utilization-status UTILIZED
```

## Keluar dari pendaftaran server game

[Ketika permainan berakhir, server game harus membatalkan pendaftaran dari Amazon GameLift FleetIQ menggunakan \(\). DeregisterGameServer](#)

```
AWS gamelift deregister-game-server \  
  --game-server-group-name MyLiveGroup \  
  --game-server-id UniqueId-1234
```

## Integrasikan Amazon GameLift FleetIQ ke dalam klien game

Topik ini menjelaskan tugas yang diperlukan untuk mempersiapkan klien game atau layanan perjudohan Anda untuk berkomunikasi dengan Amazon GameLift FleetIQ untuk memperoleh server game untuk menjadi tuan rumah sesi permainan.

Buat metode yang memungkinkan client game atau matchmaker untuk meminta sumber daya server game untuk pemain. Anda memiliki beberapa pilihan cara melakukannya:

- Minta Amazon GameLift FleetIQ memilih server game yang tersedia. Opsi ini memanfaatkan pengoptimalan Amazon GameLift FleetIQ untuk menggunakan Instans Spot berbiaya rendah dan untuk penskalaan otomatis.
- Minta semua server game yang tersedia dan pilih salah satu untuk digunakan (sering disebut sebagai "daftar dan pilih").

## Biarkan Amazon GameLift FleetIQ memilih server game

Agar Amazon GameLift FleetIQ memilih server game yang tersedia, panggil [ClaimGameServer\(\)](#) tanpa menentukan ID server game. Dalam skenario ini, Amazon GameLift FleetIQ melakukan logikanya untuk menemukan server game pada instance yang layak untuk hosting game dan dioptimalkan untuk penskalaan otomatis.

```
AWS gamelift claim-game-server \  
  --game-server-group-name MyLiveGroup
```

Menanggapi permintaan klaim, Amazon GameLift FleetIQ mengidentifikasi `GameServer` sumber daya, informasi koneksi, dan data game, yang dapat digunakan klien untuk terhubung ke server game. Status klaim server game diatur ke `CLAIMED` selama 60 detik. Baik server game atau layanan klien Anda perlu memperbarui status server game di Amazon GameLift FleetIQ setelah pemain terhubung atau gameplay dimulai. Ini memastikan bahwa Amazon GameLift FleetIQ tidak menyediakan server game ini sebagai respons terhadap permintaan berikutnya untuk kapasitas server game. Perbarui status server game dengan menelepon [UpdateGameServer\(\)](#).

```
AWS gamelift update-game-server \  
  --game-server-group-name MyLiveGroup \  
  --game-server-id UniqueId-1234 \  
  --health-check HEALTHY \  
  --utilization-status UTILIZED
```

## Pilih server game Anda sendiri

Dengan metode “daftar dan pilih”, klien game atau mak comblang Anda meminta daftar server game yang tersedia dengan menelepon [ListGameServers\(\)](#). Anda mungkin ingin menggunakan data server game untuk memberikan informasi tambahan yang dapat digunakan pemain atau matchmaker saat memilih server game. Untuk mengontrol bagaimana hasil dikembalikan, Anda dapat meminta hasil yang dilengkapi nomor halaman dan mengurutkan server game berdasarkan tanggal pendaftaran. Permintaan berikut mengembalikan 20 server game aktif dan tersedia dalam grup server game tertentu, diurutkan berdasarkan waktu pendaftaran dengan server game terbaru diutamakan.

```
AWS gamelift list-game-servers \  
  --game-server-group-name MyLiveGroup \  
  --limit 20 \  
  --sort-order DESCENDING
```



Berdasarkan daftar server game yang tersedia, klien atau layanan perijodohan memilih server game dan mengklaimnya dengan menelepon [ClaimGameServer\(\) dengan ID](#) server game tertentu. Dalam skenario ini, Amazon GameLift FleetIQ tidak menggunakan logika pengoptimalan jenis instans apa pun, seperti yang dijelaskan dalam [Logika Amazon GameLift FleetIQ](#)

```
AWS gamelift claim-game-server \  
  --game-server-group-name MyLiveGroup \  
  --game-server-id UniqueId-1234
```

# Pantau Amazon GameLift FleetIQ dengan Amazon CloudWatch

Gunakan CloudWatch metrik Amazon untuk menskalakan kapasitas instans Anda, membangun dasbor operasi, dan memicu alarm. Amazon GameLift FleetIQ sebagai solusi mandiri memancarkan serangkaian CloudWatch metrik Amazon ke akun Anda. AWS Lihat juga [Memantau Grup dan Instans Penskalaan Otomatis Anda Menggunakan Amazon CloudWatch di Panduan Pengguna Penskalaan Otomatis Amazon EC2](#).

Metrik FleetIQ tercantum di sini. Lihat informasi CloudWatch metrik Amazon lengkap untuk [GameLift metrik Amazon GameLift di Amazon](#).

Metrik	Deskripsi
<code>AvailableGameServers</code>	<p>Server game yang tersedia untuk menjalankan eksekusi game dan saat ini tidak ditempati dengan gameplay. Nomor ini mencakup server game yang telah diklaim tetapi masih dalam status tersedianya.</p> <p>Unit: Count (Jumlah)</p> <p>CloudWatchStatistik yang relevan: Jumlah</p> <p>Dimensi: GameServerGroup</p>
<code>UtilizedGameServers</code>	<p>Server game yang saat ini ditempati dengan gameplay. Nomor ini mencakup server game yang dalam status DIMANFAATKAN.</p> <p>Unit: Count (Jumlah)</p> <p>CloudWatchStatistik yang relevan: Jumlah</p> <p>Dimensi: GameServerGroup</p>
<code>DrainingAvailableGameServers</code>	<p>Server game pada instans yang dijadwalkan untuk dihentikan karena saat ini tidak mendukung gameplay. Server game ini adalah prioritas terendah</p>

Metrik	Deskripsi
	<p>untuk diklaim sebagai tanggapan atas permintaan klaim baru.</p> <p>Unit: Count (Jumlah)</p> <p>CloudWatchStatistik yang relevan: Jumlah</p> <p>Dimensi: GameServerGroup</p>
DrainingUtilizedGameServers	<p>Server game pada instans yang dijadwalkan untuk dihentikan yang saat ini men-support gameplay.</p> <p>Unit: Count (Jumlah)</p> <p>CloudWatchStatistik yang relevan: Jumlah</p> <p>Dimensi: GameServerGroup</p>
PercentUtilizedGameServers	<p>Porsi server game yang saat ini men-support eksekusi game. Metrik ini menunjukkan jumlah kapasitas server game yang saat ini digunakan. Hal ini berguna untuk mendorong kebijakan Auto Scaling yang dinamis dapat menambah dan menghapus instans untuk pertandingan dengan permintaan pemain.</p> <p>Unit: Persen</p> <p>CloudWatchStatistik yang relevan: Rata-rata, Minimum, Maksimum</p> <p>Dimensi: GameServerGroup</p>

Metrik	Deskripsi
GameServerInterruptions	<p>Server game di Instans spot yang terganggu karena ketersediaan Spot terbatas.</p> <p>Unit: Count (Jumlah)</p> <p>CloudWatchStatistik yang relevan: Jumlah</p> <p>Dimensi:GameServerGroup, InstanceType</p>
InstanceInterruptions	<p>Instans Spot yang terganggu karena ketersediaan yang terbatas.</p> <p>Unit: Count (Jumlah)</p> <p>CloudWatchStatistik yang relevan: Jumlah</p> <p>Dimensi:GameServerGroup, InstanceType</p>

# Panduan referensi Amazon GameLift FleetIQ

Bagian ini berisi dokumentasi referensi untuk digunakan dengan Amazon GameLift FleetIQ.

Topik

- [Referensi API layanan Amazon GameLift FleetIQ \(AWSSDK\)](#)

## Referensi API layanan Amazon GameLift FleetIQ (AWSSDK)

Topik ini menyediakan daftar tindakan API berbasis tugas untuk Amazon GameLift FleetIQ. API layanan Amazon GameLift FleetIQ dikemas ke dalam AWS SDK di namespace `aws.gamelift`. [Unduh AWS SDK](#) atau [lihat dokumentasi referensi Amazon GameLift API](#).

Amazon GameLift FleetIQ mengoptimalkan penggunaan Instans Spot berbiaya rendah untuk hosting game berbasis cloud dengan Amazon EC2. Lihat [Panduan GameLift Pengembang Amazon](#) untuk informasi lebih lanjut tentang opsi GameLift hosting Amazon lainnya.

## Tindakan API Amazon GameLift FleetIQ

Operasi berikut memungkinkan Anda mengelola sumber daya Amazon GameLift FleetIQ, termasuk grup server game dan server game, bersama dengan grup Amazon EC2 dan Auto Scaling.

### Mengelola grup server game

Gunakan operasi ini untuk mengelola deployment server game Anda dengan optimasi FleetIQ. Grup server game mengontrol bagaimana proses server game Anda diluncurkan pada instans Amazon EC2, pengaturan dan grup Auto Scaling, dan mendefinisikan bagaimana menerapkan optimasi FleetIQ.

- [CreateGameServerGroup](#)- Buat grup server game baru dan grup Auto Scaling yang sesuai, dan mulailah meluncurkan instance untuk meng-host server game Anda. Perintah CLI: [create-game-server-group](#)
- [ListGameServerGroups](#)- Dapatkan daftar semua grup server game di GameLift wilayah Amazon. Perintah CLI: [list-game-server-groups](#)
- [DescribeGameServerGroup](#)- Ambil metadata untuk grup server game. Perintah CLI: [describe-game-server-group](#)

- [UpdateGameServerGroup](#)- Ubah metadata grup server game. Perintah CLI: [update-game-server-group](#)
- [DeleteGameServerGroup](#)- Hapus grup server game secara permanen dan hentikan aktivitas FleetIQ untuk sumber daya hosting terkait. Perintah CLI: [delete-game-server-group](#)
- [ResumeGameServerGroup](#)- Kembalikan aktivitas FleetIQ yang ditangguhkan untuk grup server game. Perintah CLI: [resume-game-server-group](#)
- [SuspendGameServerGroup](#)- Menghentikan sementara aktivitas FleetIQ untuk grup server game. Perintah CLI: [suspend-game-server-group](#)

## Mengelola server game

Gunakan operasi ini untuk mengelola deployment server game Anda dengan optimasi FleetIQ. Grup server game mengontrol bagaimana proses server game Anda diluncurkan pada instans Amazon EC2, pengaturan dan grup Auto Scaling, dan mendefinisikan bagaimana menerapkan optimasi FleetIQ.

- [RegisterGameServer](#)- Panggilan dari server game baru untuk memberi tahu Amazon GameLift FleetIQ bahwa server game siap untuk meng-host gameplay. Perintah CLI: [register-game-server-group](#)
- [ListGameServers](#)- Panggilan dari layanan klien game untuk mendapatkan daftar semua server game yang saat ini berjalan dalam grup server game. Perintah CLI: [list-game-servers](#)
- [ClaimGameServer](#)- Panggilan dari layanan klien game untuk mencari dan memesan server game untuk menjadi tuan rumah sesi permainan baru. Perintah CLI: [claim-game-server](#)
- [DescribeGameServer](#)- Ambil metadata untuk server game. Perintah CLI: [describe-game-server](#)
- [UpdateGameServer](#)- Ubah metadata server game, status kesehatan, atau status pemanfaatan. Perintah CLI: [update-game-server](#)
- [DeregisterGameServer](#)- Panggilan dari server game terminating untuk meminta Amazon GameLift FleetIQ menghapus server game dari grup server game. Perintah CLI: [deregister-game-server](#)

## Bahasa pemrograman yang tersedia

AWSSDK dengan dukungan untuk Amazon GameLift tersedia dalam bahasa berikut. Untuk informasi tentang dukungan untuk lingkungan pengembangan, lihat dokumentasi untuk setiap bahasa.

- [C ++ \(dokumen SDK\) \(Amazon\) GameLift](#)

- [Java \(dokumen SDK\) \(Amazon\) GameLift](#)
- [.NET \(dokumen SDK\) \(Amazon\) GameLift](#)
- [Pergi \(dokumen SDK\) \(Amazon\) GameLift](#)
- [Python \(dokumen SDK\) \(Amazon\) GameLift](#)
- [Ruby \(dokumen SDK\) \(Amazon\) GameLift](#)
- [PHP \(dokumen SDK\) \(Amazon\) GameLift](#)
- [JavaScript/Node.js \(dokumen SDK\) \(Amazon\) GameLift](#)

# Keamanan dengan Amazon GameLift FleetIQ

Jika Anda menggunakan Amazon GameLift FleetIQ sebagai fitur mandiri dengan Amazon EC2, lihat Keamanan [di Amazon EC2 di Panduan Pengguna Amazon EC2](#).

Keamanan cloud di AWS adalah prioritas tertinggi. Sebagai pelanggan AWS , Anda mendapatkan manfaat dari pusat data dan arsitektur jaringan yang dibangun untuk memenuhi persyaratan dari organisasi yang paling sensitif terhadap keamanan.

Keamanan adalah tanggung jawab bersama antara Anda AWS dan Anda. Untuk informasi tentang cara menerapkan model tanggung jawab bersama saat menggunakan Amazon GameLift FleetIQ, lihat [Keamanan](#) di Amazon. GameLift



# Catatan rilis Amazon GameLift FleetIQ dan versi SDK

Catatan GameLift rilis Amazon memberikan detail tentang fitur, pembaruan, dan perbaikan FleetIQ baru yang terkait dengan layanan. Halaman ini juga mencakup riwayat versi Amazon GameLift SDK.

# Sumber daya GameLift pengembang Amazon

Untuk melihat semua GameLift dokumentasi Amazon dan sumber daya pengembang, lihat halaman beranda [GameLiftDokumentasi Amazon](#).

# AWSGlosarium

Untuk AWS terminologi terbaru, lihat [AWSglosarium di Referensi](#). Glosarium AWS

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