亚马逊云科技

FleetIQ 开发人员指南

Amazon GameLift Servers

版本

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

Amazon GameLift Servers: FleetIQ 开发人员指南

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

Amazon 的商标和商业外观不得用于任何非 Amazon 的商品或服务，也不得以任何可能引起客户混淆、贬低或诋毁 Amazon 的方式使用。所有非 Amazon 拥有的其他商标均为各自所有者的财产，这些所有者可能附属于 Amazon、与 Amazon 有关联或由 Amazon 赞助，也可能不是如此。

Amazon Web Services 文档中描述的 Amazon Web Services 服务或功能可能因区域而异。要查看适用于中国区域的差异，请参阅 中国的 Amazon Web Services 服务入门 (PDF)。

Table of Contents

什么是 Amazon GameLift Servers FleetIQ? 1
- 操作方法 FleetIQ 工作 1
  - Amazon GameLift Servers FleetIQ 逻辑 2
  - 关键资源和组件 5
- 游戏架构 6
  - 补充本地托管 6
- 游戏服务器组的生命周期 8
- 游戏服务器的生命周期 10
- 竞价平衡过程 12
- 最佳实践 14
- Amazon GameLift Servers FleetIQ 功能 16
- 的定价 Amazon GameLift ServersFleetIQ 17

设置 18
- 支持的软件 18
- 设置您的 Amazon 账户 19
  - 创建一个 Amazon Web Services 账户 19
  - 管理用户权限 Amazon GameLift Servers FleetIQ 20
  - 为跨服务交互创建 IAM 角色 24

正在为游戏做准备 FleetIQ 30
- 集成步骤 30
- 管理游戏服务器组 32
  - 创建游戏服务器组 32
  - 更新游戏服务器组 33
  - 跟踪游戏服务器组实例 33
- 集成游戏服务器 34
  - 注册游戏服务器 34
  - 更新游戏服务器状态 34
  - 注销游戏服务器 35
- 集成游戏客户端 35
  - 让 Amazon GameLift Servers FleetIQ 选择游戏服务器 36
  - 选择您自己的游戏服务器 36

使用监视器 CloudWatch 38

安全性与 FleetIQ 40

Amazon GameLift Servers FleetIQ 参考 41

服务 API 参考 (Amazon SDK) 41
Amazon GameLift Servers FleetIQ API 操作 41
可用编程语言 42
发行说明和软件开发工具包版本 43
全部 Amazon GameLift Servers 导游 43
Amazon 词汇表 44
..... xlv

什么是 Amazon GameLift Servers FleetIQ?

Amazon GameLift Servers FleetIQ 优化了使用低成本的亚马逊弹性计算云 (Amazon EC2) 竞价型实例进行基于云的游戏托管。With Amazon GameLift Servers FleetIQ，您可以直接使用亚马逊 EC2 和 Amazon A EC2 uto Scaling 中的托管资源，同时充分利用 Amazon GameLift Servers 优化，为您的玩家提供价格低廉、弹性强的游戏托管。Amazon EC2 Spot 实例虽然以大幅折扣提供，但通常不适用于游戏托管，因为可用性会波动，而且有可能出现中断。Amazon GameLift Servers FleetIQ 显著缓解了这些限制，使得使用低成本竞价型实例可以用于游戏托管。

FleetIQ 使用时也可以进行优化 Amazon GameLift Servers 来管理您的游戏托管。有关信息 Amazon GameLift Servers 托管选项，请参阅 Amazon GameLift Servers 开发者指南。

这些区域有：Amazon GameLift Servers FleetIQ 游戏托管解决方案专为符合以下条件的游戏开发者而设计：

- 已有 Amazon 部署或想 EC2 直接使用 Amazon 而不是通过完全托管方式使用 Amazon Amazon GameLift Servers 服务。Amazon GameLift Servers FleetIQ 与您在中管理的 EC2 Auto Scaling 群组配合使用 Amazon Web Services 账户，使您能够完全访问自己的 EC2 实例和群组。您还可以与其他 Amazon 服务集成，包括亚马逊弹性容器服务 (Amazon ECS)、亚马逊 Elastic Kubernetes Service（亚马逊 EKS）和。 Amazon Shield Advanced
- 拥有现有的本地游戏托管，并希望将容量扩展到云。With Amazon GameLift Servers FleetIQ，您可以构建一个混合部署系统，该系统使用您的本地容量，并根据需要逐步增加 Amazon 云容量。

> 准备好开始工作了 Amazon GameLift Servers FleetIQ?
>
> - 学习如何使用 Amazon GameLift Servers FleetIQ 通过使用Amazon参加课程来为你的游戏增光添彩 Amazon GameLift Servers FleetIQ 适用于 Amazon 技能生成器上的游戏服务器。有关相关课程的概述，请参阅游戏技术学习计划。有些课程有不同的语言版本。
> - 按照Amazon GameLift Servers FleetIQ 集成步骤中的说明进行操作。

操作方法 Amazon GameLift Servers FleetIQ 工作

这些区域有：Amazon GameLift Servers FleetIQ 解决方案是一个游戏托管层，它补充了你通过 Amazon EC2 和 Auto Scaling 获得的全套计算资源管理工具。除了提供一系列特定于游戏托管的功能

外，Amazon GameLift Servers FleetIQ 提供了额外的逻辑层，使得使用低成本竞价型实例进行游戏托管成为可能。该解决方案允许您直接管理您的 Amazon EC2 和 Auto Scaling 资源，并根据需要与其他 Amazon 服务集成。

使用时 Amazon GameLift Servers FleetIQ，您要像往常一样准备启动亚马逊 EC2 实例：使用游戏服务器软件制作亚马逊系统映像 (AMI)，创建亚马逊 EC2 启动模板，然后为 Auto Scaling 组定义配置设置。但是，与其直接创建 Auto Scaling 组，不如创建一个 Amazon GameLift Servers FleetIQ 包含您的 Amazon EC2 和 Auto Scaling 资源和配置的游戏服务器组。此操作会提示 Amazon GameLift Servers FleetIQ 创建游戏服务器组和相应的 Auto Scaling 组。游戏服务器组与自动扩缩组关联并管理自动扩缩组的某些方面。

创建 Auto Scaling 组后，您可以完全访问您的 Amazon EC2 和 Auto Scaling 资源。您可以更改 Auto Scaling 组的配置，添加多级扩展策略或负载均衡器，以及与其他 Amazon 服务集成。您可以直接连接到组中的实例。作为其优化逻辑的一部分，Amazon GameLift Servers FleetIQ 还会定期更新某些 Auto Scaling 组属性。您可以跟踪自动扩缩组部署的所有实例的可用性状态。

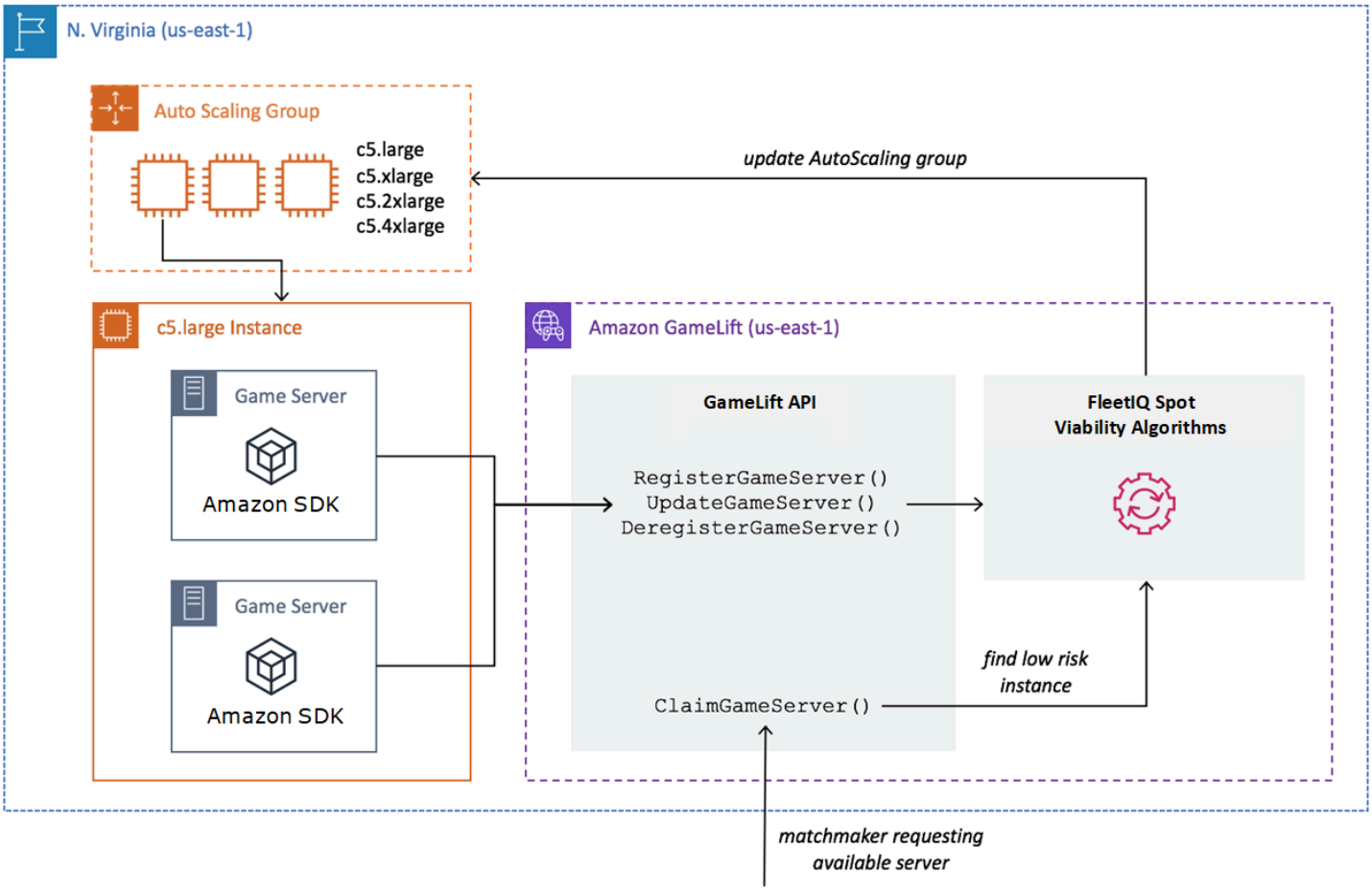
你可以暂时暂停 Amazon GameLift Servers FleetIQ 游戏服务器组随时的活动。您还可以选择删除游戏服务器组，但保留对应的自动扩缩组。

主题

- Amazon GameLift Servers FleetIQ 逻辑
- 关键资源和组件

Amazon GameLift Servers FleetIQ 逻辑

下图说明了的角色 Amazon GameLift Servers FleetIQ 当它与亚马逊合作托 EC2管游戏时。其主要目标是找到最好 的游戏服务器来托管游戏会话，并为玩家提供最佳的游戏体验。Amazon GameLift Servers FleetIQ 将最佳资源定义为那些以最低成本提供最高游戏托管可行性的资源。Amazon GameLift Servers FleetIQ 通过两种关键方式实现这一目标：第一，在 Auto Scaling 组中仅允许可行的实例类型，其次是在该组的可用资源中有效地放置新的游戏会话。



使用最佳实例类型填充自动扩缩组

自动扩缩组的任务是启动新实例并停用旧实例，同时维护托管资源集合并扩展该集合以满足玩家需求。为此，自动扩缩组依赖于您所需的实例类型列表。的工作 Amazon GameLift Servers FleetIQ 就是不断检查这些所需实例类型的可行性，并更新 Auto Scaling 组的列表。此过程称为实例平衡。它可确保自动扩缩组中的实例不断刷新，以便始终仅使用当前可行的实例类型。

Amazon GameLift Servers FleetIQ 影响 Auto Scaling 组如何通过以下方式选择最佳实例类型：

- 它决定竞价型和/或按需型实例的使用情况。网络 ACL 和安全组都允许 (因此可到达您的实例) 的发起 ping 的 Amazon GameLift Servers FleetIQ 游戏服务器组配置了平衡策略，这会影响 Auto Scaling 组使用竞价型和/或按需实例的方式。由于可用性波动和潜在的中断，竞价型实例的成本较低，这些限制包括 Amazon GameLift Servers FleetIQ 最小化游戏服务器托管。按需型实例成本更高昂，但在您需要时可提供更可靠的可用性。
- 它限制新实例只能在可行的实例类型上启动。A Amazon GameLift Servers FleetIQ 游戏服务器组维护着所需实例类型的主列表。实例平衡过程使用预测算法，持续评估列表中每种所需的实例类型，以确定游戏托管的可行性，该算法会查看该实例类型的最新可用性和中断率。作为这次评估的结

果,Amazon GameLift Servers FleetIQ 不断更新 Auto Scaling 组的所需实例类型列表，使其仅包括当前可行的实例类型。

- 它会标记不可行的实例类型的现有实例。Amazon GameLift Servers FleetIQ 标识 Auto Scaling 组中当前不可行的实例类型的现有实例。这些实例标记为耗尽，这意味着将终止这些实例并使用新实例替换。对于已启用游戏服务器保护的实例，终止将推迟到任何活动的游戏会话正常结束。

随着自动扩缩组启动和停用实例，它将持续维护针对游戏托管进行优化的集合，即使低成本竞价型实例类型的可用性不断波动也是如此。平衡活动仅在具有活动实例的游戏服务器组上进行。在竞价平衡过程中了解有关此过程如何工作的详细信息。

有效地放置游戏会话

Amazon GameLift Servers FleetIQ 跟踪游戏服务器组中所有活跃的游戏服务器，并使用此信息来确定新游戏会话和玩家的最佳位置。

要启用 Amazon GameLift Servers FleetIQ 要跟踪游戏服务器，您的游戏服务器软件必须报告其状态。您的自定义 AMI 控制在每个实例上启动和停止新游戏服务器进程的方式。当新的游戏服务器启动时，它会注册到 Amazon GameLift Servers FleetIQ，表示它已准备好主持游戏会话。在注册后，游戏服务器定期报告其运行状况以及当前是否托管游戏会话。当游戏服务器关闭时，它会注销注册到 Amazon GameLift Servers FleetIQ.

要开始新的游戏会话，您的游戏客户端（或媒人或其他客户端服务）会将游戏服务器请求发送到 Amazon GameLift Servers FleetIQ. Amazon GameLift Servers FleetIQ 找到可用的游戏服务器，为新游戏会话申领该服务器，然后使用游戏服务器 ID 和连接信息进行响应。然后，游戏会提示游戏服务器更新其状态，并为进入的玩家启动新的游戏会话。

在选择游戏服务器来托管新的游戏会话时，Amazon GameLift Servers FleetIQ 使用以下决策过程来优化可行的低成本 Spot 实例的放置：

1. 在可能的情况下，Amazon GameLift Servers FleetIQ 在已经托管其他游戏会话的实例上放置新的游戏会话。通过打包（但不过载）某些实例并保持其他实例为空闲状态，自动扩缩组能够在不需要空闲实例时快速缩减空闲实例，从而降低托管成本。
2. Amazon GameLift Servers FleetIQ 忽略被标记为耗尽的实例，即不适用于游戏托管的实例。这些实例仅为持现有的游戏会话而保持运行。除非没有其他游戏服务器可用，否则它们不能用于新的游戏会话。
3. Amazon GameLift Servers FleetIQ 标识在可行实例上运行的所有可用游戏服务器。

您可以为游戏服务器组启用游戏会话保护，以防止自动扩缩组终止具有正在运行游戏会话的实例。

关键资源和组件

在设置游戏托管资源之前，请在 Amazon 账户中创建以下资源 Amazon GameLift Servers FleetIQ。作为最佳实践，在通过游戏服务器组使用这些资源之前，请先使用这些资源开发和测试游戏服务器部署。

- 亚马逊机器映像（AMI）。AMI 是您想要通过 Amazon EC2 实例启动的特定软件配置的模板。对于游戏托管，AMI 包括操作系统、游戏服务器二进制文件或容器，以及游戏服务器所需的其他运行时软件。有关创建 AMI 的更多信息，请参阅亚马逊 EC2 用户指南中的亚马逊系统映像。 AMIs 是特定于区域的。您可以按照 Amazon EC2 用户指南中的复制 AMIs中所述，将 AMI 从一个区域复制到另一个区域。
- 亚马逊 EC2 发布模板。启动模板提供了在自动扩缩组中启动和管理实例的说明。它指定 AMI，提供合适的实例类型列表，设置网络、安全性和其他属性。有关创建启动模板的更多信息，请参阅 Amazon EC2 用户指南中的从启动模板启动实例。启动模板特定于区域。
- Amazon IAM 角色。IAM 角色定义了一组权限，允许对 Amazon 资源进行有限访问。可信实体（例如其他 Amazon 服务）可以担任该角色并继承其权限。使用时 Amazon GameLift Servers FleetIQ，您必须为 IAM 角色提供托管策略，该策略允许 Amazon GameLift Servers FleetIQ 在您的 Amazon 账户中创建和访问 Auto Scaling 组和 EC2 实例资源。IAM 角色不是特定于区域的。

Amazon GameLift Servers FleetIQ 直接管理以下资源并对其拥有直接权限。

- Amazon GameLift Servers 游戏服务器组。游戏服务器组包含用于定义如何使用的配置设置 Amazon GameLift Servers FleetIQ 与相应的 Auto Scaling 小组合作，提供低成本的游戏托管。游戏服务器组是区域特定的。当您在某个区域中创建游戏服务器组时，系统会在同一区域的 Amazon 账户中自动创建一个新的 Auto Scaling 组。游戏服务器组与 Auto Scaling 组关联，并且有权（通过担任 IAM 角色）管理和修改其部分设置。游戏服务器组是一个长寿命的资源；开发人员应预期很少创建它们。游戏服务器组也是托管在 Auto Scaling 组中的实例上并注册到的游戏服务器的功能分组资源 Amazon GameLift Servers FleetIQ.
- Amazon GameLift Servers 游戏服务器。 游戏服务器资源表示在与之关联的实例上运行的游戏执行 Amazon GameLift Servers FleetIQ 游戏服务器组。此资源是在游戏服务器向注册时创建的 Amazon GameLift Servers FleetIQ 并标识它所属的游戏服务器组。Amazon GameLift Servers FleetIQ 跟踪每台已注册游戏服务器的使用状态和申领状态，从而使其能够监控游戏服务器的可用性。游戏服务器特定于区域，因为它们与特定于区域的游戏服务器组相关联。当您的游戏请求新的游戏服务器时，它会指定游戏服务器组和区域。

这些资源是通过以下方式创建的 Amazon GameLift Servers FleetIQ 资源的费用。它们在您的 Amazon 账户中创建，您对它们具有完全控制权限。

- Amazon A EC2 uto Scaling 小组。Auto Scaling 组启动和管理一组 EC2 实例，并自动扩展组容量。With Amazon GameLift Servers FleetIQ，游戏服务器组和 Auto Scaling 组之间存在 one-to-one 关系。虽然您可以更新 Auto Scaling 组的所有设置，Amazon GameLift Servers FleetIQ 作为其逻辑的一部分，定期覆盖和更新某些设置，以平衡竞价型实例的游戏托管可行性。有关更多信息，请参阅 Amazon A EC2 uto Scaling 用户指南 AutoScalingGroup中的。Auto Scaling 组是特定于区域的；它们创建在与游戏服务器组相同的区域中。
- 亚马逊 EC2 实例。实例是云中的虚拟服务器。实例类型具有指定计算、内存、磁盘和网络资源的特定硬件配置。它们通常由具有 AMI 的自动扩缩组启动。实例可以是 Spot 或按需实例，具体取决于可用性。With Amazon GameLift Servers FleetIQ，实例运行一个或多个游戏服务器进程，每个进程都可以托管多个游戏会话。实例特定于区域，因为它们与特定于区域的自动扩缩组相关联。

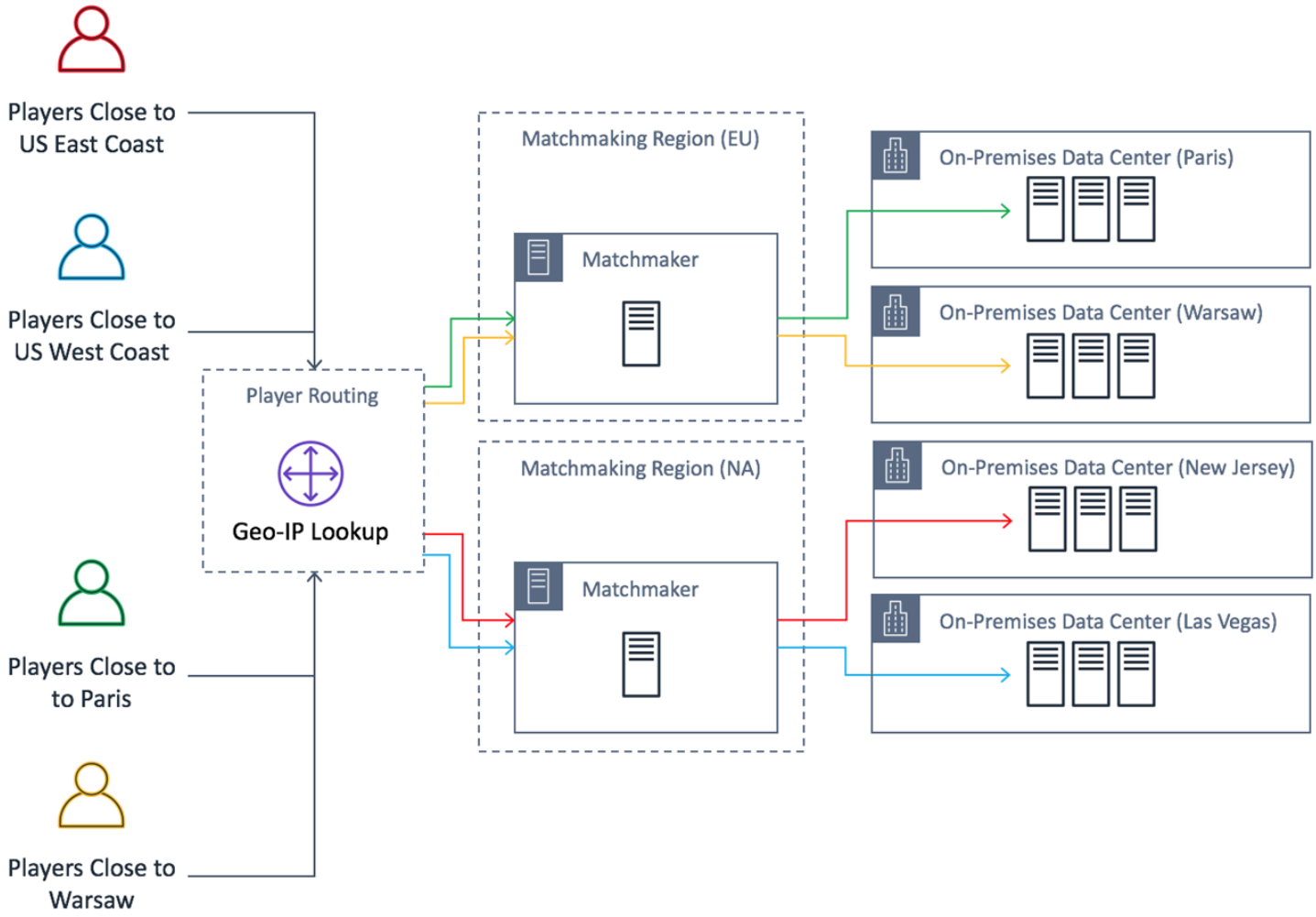
游戏架构与 Amazon GameLift Servers FleetIQ

补充本地托管

Amazon GameLift Servers FleetIQ 旨在重复使用您现有的游戏后端，包括您可能已经拥有的任何玩家地理IP路由、配对或大厅服务。以下示例说明了如何 Amazon GameLift Servers FleetIQ 可能适合现有的本地部署。

Example

在此示例中，游戏托管最初由四个专有数据中心处理，以便在北美和欧洲托管玩家。根据它们的大致实际位置，玩家会被路由到两个区域匹配器中的一个。匹配器会按技能和延迟对玩家进行分组，然后将他们放到附近的游戏服务器上，以最大限度地减少滞后。

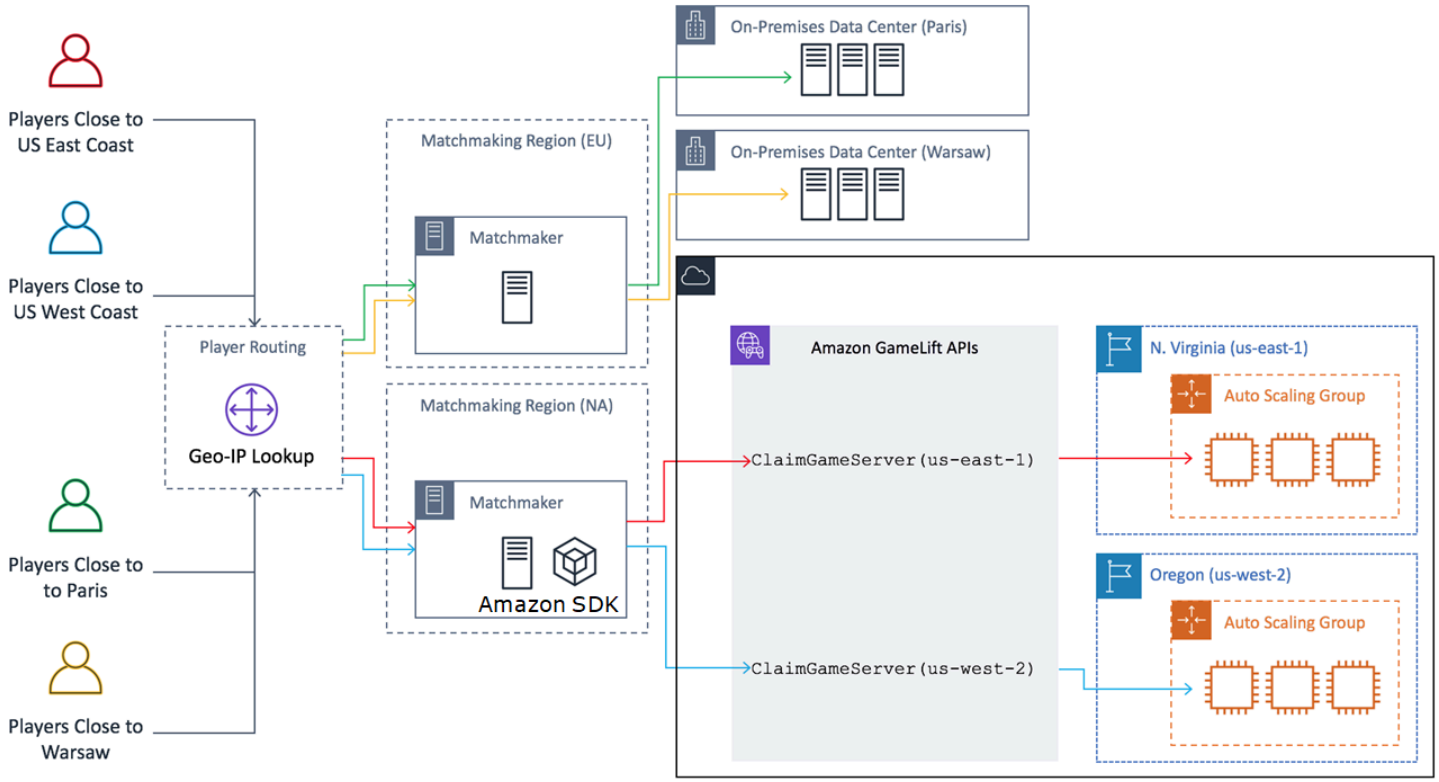


游戏开发者想用由提供的服务器替换其北美游戏服务器 Amazon GameLift Servers FleetIQ。 首先，他们对游戏服务器进行了细微的更新，使其能够与之配合使用 Amazon GameLift Servers FleetIQ 然后创建亚马逊系统映像 (AMI)。此镜像将安装在为游戏部署的每个 EC2 实例上。该映像包含游戏服务器、依赖关系以及为玩家运行游戏会话所需的任何其他内容。

AMI 准备就绪后，开发者创建了两个 Amazon GameLift Servers FleetIQ 游戏服务器组，每个 Amazon 北美地区一个`us-east-1`（`us-west-2`)和. 开发者传入启动模板（提供 AMI）、所需实例类型列表以及该组的其他配置设置。所需的实例类型列表告诉我们 Amazon GameLift Servers FleetIQ 在检查是否有适用于游戏托管的竞价型实例时要使用哪些类型。

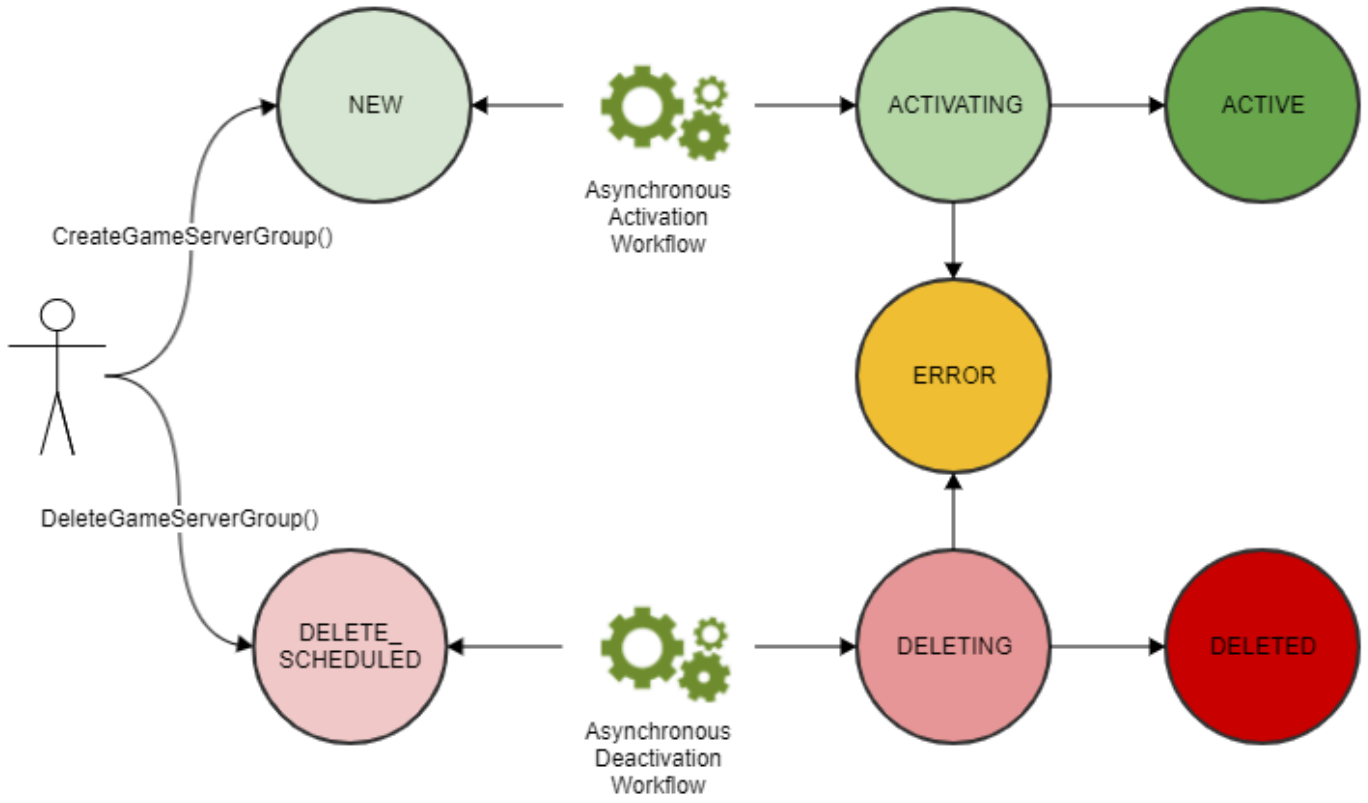
最后，开发人员将 Amazon SDK 与 Amazon GameLift Servers FleetIQ 进入他们的北美媒人，它打电话给 Amazon GameLift Servers FleetIQ 当一组新的玩家需要服务器容量来进行游戏会话时。Amazon GameLift Servers FleetIQ 找到带有可用游戏服务器的竞价型实例，将其保留给玩家，并提供服务器连接信息。玩家连接到服务器，玩游戏，然后断开连接。要开始新游戏，玩家需要重新进入配对，这会提示 Amazon GameLift Servers FleetIQ 寻找其他可用的游戏服务器。每个新的游戏请求都会触发

Amazon GameLift Servers FleetIQ 搜索和选择中断几率较低的游戏服务器。结果，Amazon GameLift Servers FleetIQ 会不断将玩家从无法托管游戏的游戏服务器上重定向，即使竞价型实例的可用性会随着时间的推移而波动。



游戏服务器组的生命周期

游戏服务器组经历以下生命周期，包括预配置和状态更新。游戏服务器组预计是一个长寿命的资源。



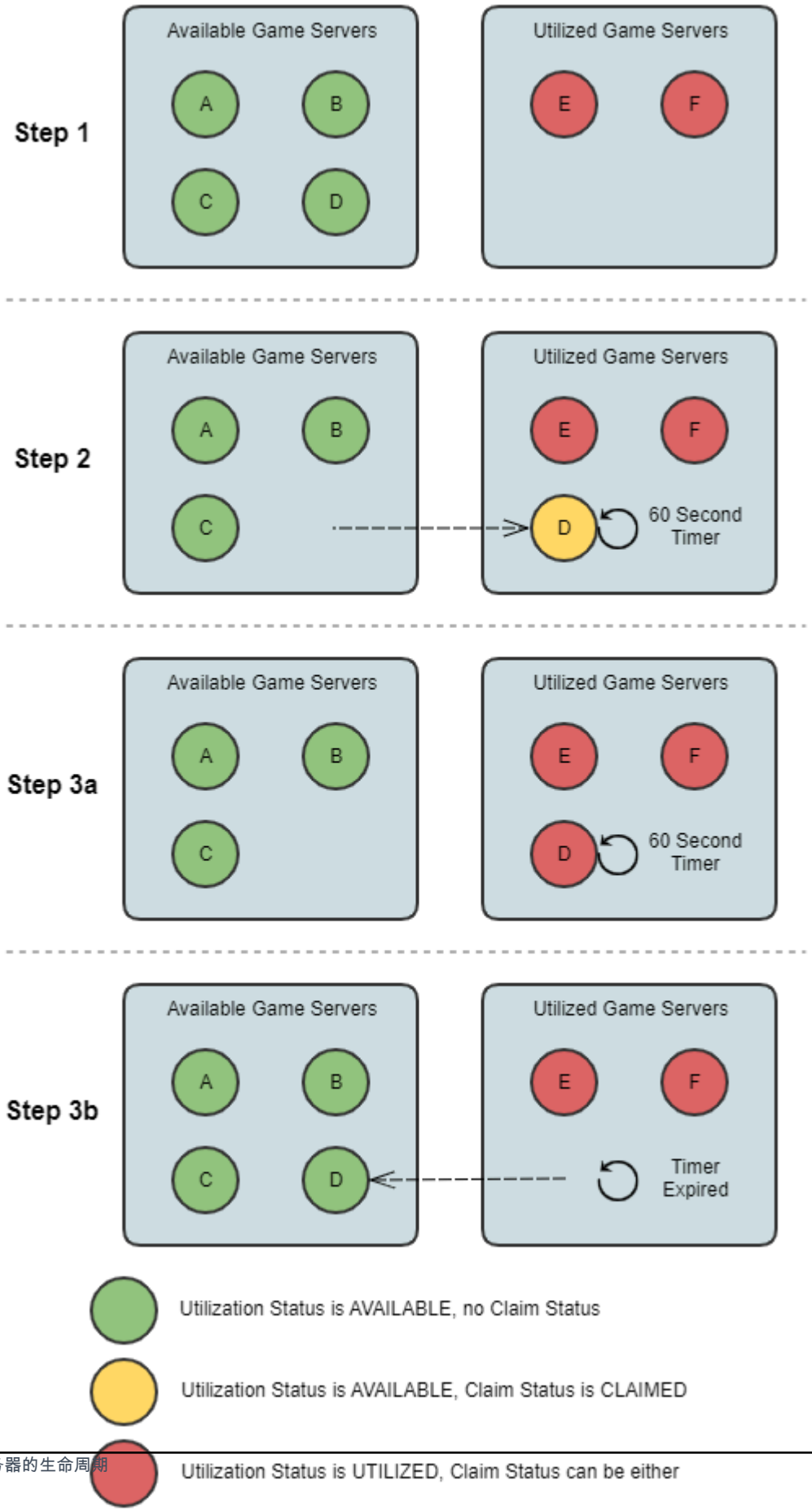
- 要创建游戏服务器组，请调用 Amazon GameLift Servers API `CreateGameServerGroup()` 并传入 EC2 启动模板和配置设置。为响应呼叫，将创建一个新的游戏服务器组并将其置于 NEW（新）状态。
- Amazon GameLift Servers FleetIQ 激活异步激活工作流程，将游戏服务器组状态转换为 “激活”。该工作流程启动底层资源的创建，包括 Amazon A EC2 uto Scaling 组和具有所提供 AMI 的 EC2 实例。
  - 如果预配置因任何原因失败，游戏服务器组将被置于 ERROR（错误）状态。要获取其他错误信息以帮助调试故障原因，请在处于错误状态的游戏服务器组上调用 `DescribeGameServerGroup()`。
  - 如果预配置成功，则游戏服务器组将转换为 ACTIVE（活动）状态。此时，将使用注册到的游戏服务器启动实例 Amazon GameLift Servers FleetIQ。 定期评估该组的实例类型以确定游戏托管的可行性，并根据需要进行平衡。Amazon GameLift Servers FleetIQ 还会跟踪组中活动游戏服务器的状态并响应对游戏服务器的请求。
- 可以通过使用组标识符调用 `DeleteGameServerGroup()` 来删除游戏服务器组。此操作将游戏服务器组置于 DELETE_SCHEDULED 状态。只能计划删除处于 ACTIVE（活动”）或 ERROR（错误）状态的游戏服务器组。

- Amazon GameLift Servers FleetIQ 激活异步停用工作流程以响应 DELETE_SCEDULED 状态，将游戏服务器组状态转换为 DELETE_SCEDULE。您可以选择仅删除游戏服务器组，或者同时删除游戏服务器组和关联的自动扩缩组。
  - 如果停用因任何原因失败，游戏服务器组将进入 ERROR（错误）状态。要获取其他错误信息以帮助调试故障原因，请在处于错误状态的游戏服务器组上调用 `DescribeGameServerGroup()`。
  - 如果停用成功，游戏服务器组将转换为 DELETED（已删除）状态。

游戏服务器的生命周期

With Amazon GameLift Servers FleetIQ，游戏服务器会经历以下生命周期，包括配置和状态更新。游戏服务器预计是短寿命的资源。最佳实操是，游戏服务器应在游戏会话结束后注销，而不是重新用于其他游戏会话。这种方法有助于确保可用的游戏服务器始终在可用于游戏托管的成本最低的资源上运行。

- 游戏服务器资源是在游戏服务器处理时创建的，在实例上运行 Amazon GameLift Servers FleetIQ关联的 Auto Scaling 群组，调用 Amazon GameLift Servers `RegisterGameServer()`要通知的 API Amazon GameLift Servers FleetIQ 它已经准备好接待玩家和游戏玩法了。游戏服务器具有两种状态来跟踪其当前可用性：
  - 利用状态跟踪游戏服务器当前是否支持玩游戏。此状态最初设置为 AVAILABLE（可用），表示它已准备好接受新的游戏。一旦游戏服务器被游戏占用，此状态就会设置为 UTILIZED（已利用）。
  - 认领状态可跟踪游戏服务器是否被认领用于即将开始的游戏。处于 CLAIMED（已认领）状态的游戏服务器表示它已被游戏客户端（或游戏服务，如匹配器）暂时保留。此状态可防止 Amazon GameLift Servers FleetIQ 从向多个请求者提供相同的游戏服务器。认领状态为空的游戏服务器可供认领。
- 下图说明游戏服务器的利用状态和认领状态在其生命周期内如何变化。


Step 1
Available Game Servers
A
B
C
D
Utilized Game Servers
E
F
Step 2
Available Game Servers
A
B
C
Utilized Game Servers
E
F
D
60 Second Timer
Step 3a
Available Game Servers
A
B
C
Utilized Game Servers
E
F
D
60 Second Timer
Step 3b
Available Game Servers
A
B
C
D
Utilized Game Servers
E
F
Timer Expired
Utilization Status is AVAILABLE, no Claim Status
Utilization Status is AVAILABLE, Claim Status is CLAIMED
Utilization Status is UTILIZED, Claim Status can be either

- 第 1 步：一个游戏服务器组有六个已注册的游戏服务器。其中四个的利用状态为 AVAILABLE（A、B、C 和 D），两个目前处于 UTILIZED 状态（E 和 F）。
- 第 2 步：游戏客户端或配对系统调用 Amazon GameLift Servers `ClaimGameServer()`用于请求新游戏服务器的 API。此请求提示 Amazon GameLift Servers FleetIQ 搜索可用的游戏服务器 (D) 并将其申领状态设置为 “已申领”，持续 60 秒。Amazon GameLift Servers FleetIQ 使用游戏服务器的连接信息（IP 地址和端口）以及其他可选的游戏特定数据来响应其请求。由于尚未在游戏服务器上开始玩游戏，因此它的利用状态仍然保持为 AVAILABLE，但无法通过其他请求来认领。
- 步骤 3a. 使用提供的连接信息，游戏客户端可以连接到游戏服务器并启动游戏。必须在 60 秒内触发游戏服务器 (D) 才能将其使用状态更改为 “已利用”，方法是调用 Amazon GameLift Servers API `UpdateGameServer()`。
- 步骤 3b. 如果游戏服务器的利用状态在 60 秒后未更新，认领计时器将过期，认领状态将重置为空白。游戏服务器 (D) 将返回到可用和未认领的游戏服务器池中。

- 游戏服务器上的游戏完成并且玩家断开连接后，游戏服务器资源将被移除。在关闭之前，游戏服务器进程会调用 Amazon GameLift Servers `DeregisterGameServer()`要通知的 API Amazon GameLift Servers FleetIQ 它脱离了游戏服务器组的游戏服务器池。

竞价平衡过程

Amazon GameLift Servers FleetIQ 定期平衡具有竞价型实例的 Auto Scaling 组中的实例。对于使用 ON_DEMAND_ONLY 平衡策略或没有任何活动实例的游戏服务器组，此过程不处于活动状态。

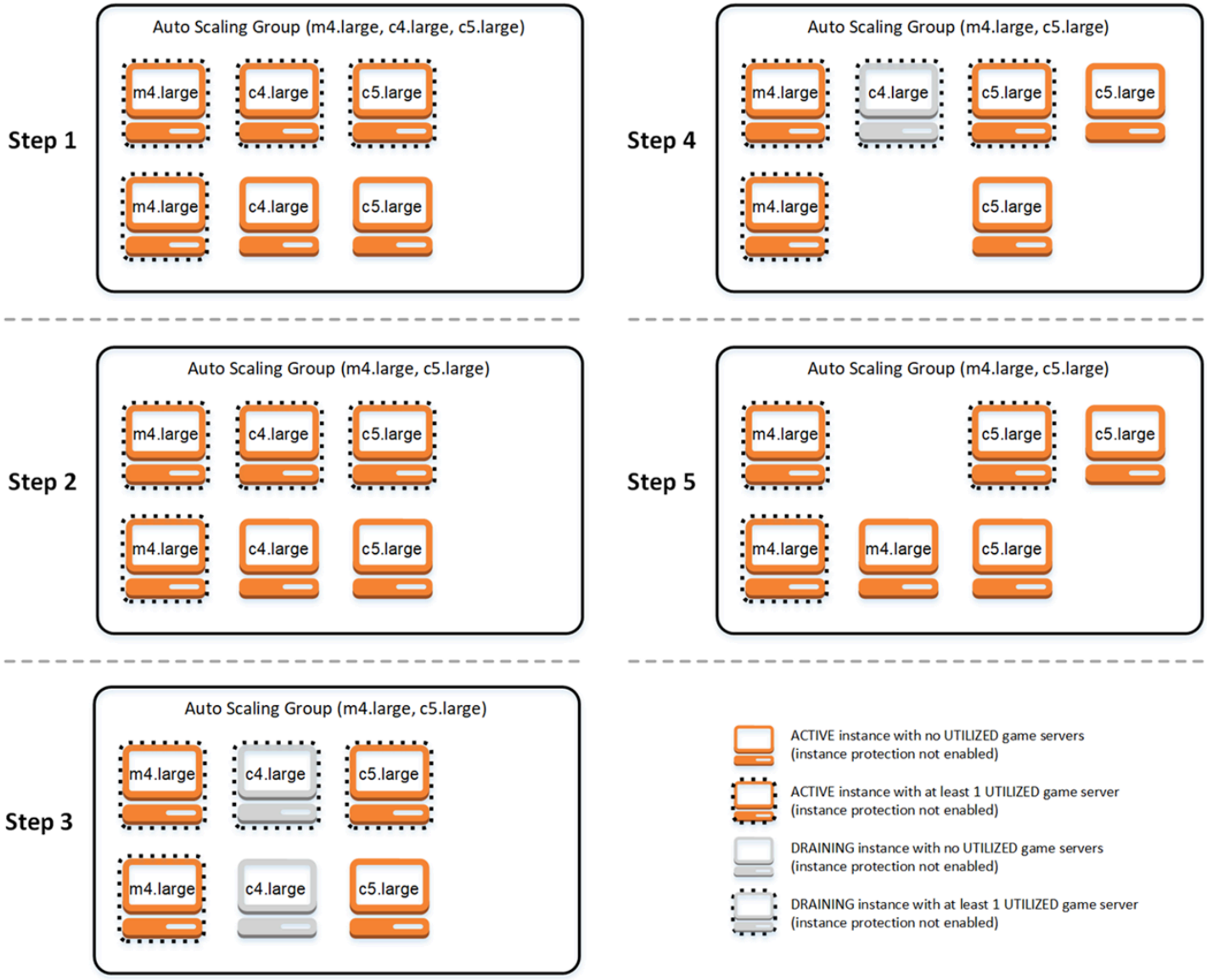
竞价平衡有两个关键目标：

- 仅使用对游戏托管具有高可行性的竞价型实例类型持续刷新组。
- 使用多种可行的实例类型（在可能时），以减少游戏服务器意外中断的影响

Amazon GameLift Servers FleetIQ 通过评估组的实例类型并移除更有可能导致游戏服务器中断的实例来实现平衡。为了避免在平衡过程中终止具有活动游戏的实例，最好是为生产中的游戏服务器组启用游戏服务器保护。

Example

以下示例说明自动扩缩组中的实例如何受到竞价平衡的影响。



- 第 1 步：通过游戏服务器组，关联的自动扩缩组设置为在启用游戏服务器保护的情况下启动类型为 m4.large、c4.large 和 c5.large 的实例。自动扩缩组已启动一个平衡集合，其中包含每种类型的两个竞价型实例。四个实例至少有一个处于 UTILIZED（被利用）状态的游戏服务器（显示为虚线边框），而两个实例当前不支持玩游戏。
- 第 2 步：Amazon GameLift Servers FleetIQ 评估所有三种实例类型的当前游戏托管可行性。评估确定 c4.large 实例类型存在不可接受的游戏服务器中断的可能性。Amazon GameLift Servers FleetIQ 立即更新 Auto Scaling 组配置，暂时将 c4.large 从实例类型列表中删除，从而防止启动其他 c4.large 实例。
- 第 3 步：Amazon GameLift Servers FleetIQ 识别 c4.large 类型的现有实例，并采取措施将其从组中删除。作为第一步，在 c4.large 实例上运行的所有游戏服务器都标记为耗尽。只有在没有其他游戏

服务器可用的情况下，才能作为最后手段认领耗尽实例上的游戏服务器。此外，将触发具有耗尽实例的自动扩缩组以启动新实例来替换它们。

- 第 4 步：随着新的可行实例上线，自动扩缩组将终止耗尽的实例。这种替换可确保该组的所需容量得到保持。要终止的第一个实例是没有已利用的游戏服务器且关闭了游戏服务器保护的 c4.large 实例。它被替换为一个新的 c5.large 实例。
- 第 5 步。对于具有游戏服务器保护的耗尽实例而言，在其游戏服务器支持玩游戏时，这些实例将继续运行。游戏结束后，一旦新的 m4.large 实例启动以取代剩余的 c5.large 实例的位置，后者就会终止。

由于此过程，自动扩缩组将保持其所需容量，同时此组将从使用三种实例类型平衡为使用两种。Amazon GameLift Servers FleetIQ 继续评估原始的实例类型列表，以确定游戏托管的可行性。当 c4.large 再次被视为可行的实例类型时，将更新自动扩缩组以包含全部三种实例类型。该组随着时间推移自然地平衡。

Amazon GameLift Servers FleetIQ 最佳实践

Amazon GameLift Servers FleetIQ 是一个低级逻辑层，可帮助您管理用于游戏托管的 Amazon EC2 资源。特别是，Amazon GameLift Servers FleetIQ 通过最大限度地减少游戏会话中断的可能性，优化了适用于游戏托管的竞价型实例的使用。它还提供基本的游戏托管功能，以跟踪可用的游戏服务器，并将游戏路由到低成本、高可行性的游戏服务器。

Amazon GameLift Servers FleetIQ 作为一项独立功能，不提供完全托管版本所提供的高级功能 Amazon GameLift Servers 解决方案，它还使用 FleetIQ 以最大限度地降低托管成本。如果您需要配对、基于延迟的玩家路由、游戏会话和玩家会话管理以及版本控制等功能，请查看 Amazon GameLift Servers 解决方案。

以下是一些最佳实践，可以帮助您从中获得最大收益 Amazon GameLift Servers FleetIQ.

- 使用 Amazon GameLift Servers FleetIQ 用于基于会话的游戏。 Amazon GameLift Servers FleetIQ 当它不断将玩家引导到不太可能出现游戏会话中断的实例时，效果最好。保持长时间的会话会干扰 Amazon GameLift Servers FleetIQ 平衡过程，这增加了游戏会话中断的可能性。理想的工作流程是让玩家从对战（或服务器选择）到进入游戏。游戏结束后，玩家会返回到对战，并路由到新实例上的另一个游戏服务器。我们建议使用 Amazon GameLift Servers FleetIQ 适用于会话时间少于两小时的游戏。
- 提供多种实例类型以供选择。设置游戏服务器组时，您提供要使用的实例类型列表。您包含的实例类型越多，灵活性就越大 Amazon GameLift Servers FleetIQ 必须使用具有

高可行性的竞价型实例进行游戏托管。例如，您可以列出同一个实例系列中的多种大小（c5.large、c5.xlarge、c5.2xlarge、c5.4xlarge）。对于大型实例，您可以在每个实例上运行更多的游戏服务器，从而可能降低成本。对于小型实例，自动缩放可以更快地响应玩家需求的变化。请记住，所需实例类型列表未按优先顺序排列，自动扩缩组将使用可行实例类型的平衡来保持该组的弹性。

- 在所有实例类型上测试您的游戏。确保游戏服务器在为游戏服务器组配置的每个实例类型上正常运行。
- 使用实例容量加权。如果您将游戏服务器组配置为使用某个实例大小范围（如 c5.2xlarge、c5.4xlarge、c5.12xlarge），请包括每种实例类型的容量加权信息。有关更多信息，请参阅 Amazon Auto Scalin g 用户指南中的 Amazon A EC2 ut EC2 o Sc aling 的实例权重。
- 使用放置游戏会话 Amazon GameLift Servers FleetIQ。 将一组玩家放置在游戏服务器上时，请使用 Amazon GameLift Servers API `ClaimGameServer()`。Amazon GameLift Servers FleetIQ 避免将玩家置于游戏会话中断几率更高的实例上。
- 将游戏服务器状态报告给 Amazon GameLift Servers FleetIQ。 定期使用报告服务器运行状况和利用率状态 Amazon GameLift Servers API `UpdateGameServer()`。保持准确的游戏服务器状态有帮助 Amazon GameLift Servers FleetIQ 更有效地放置游戏玩法。它还有助于避免在竞价平衡活动期间终止具有活动游戏的实例。
- 设置自动扩展策略。您可以创建目标跟踪扩展策略，以根据玩家利用情况和预期需求维护您的主机容量。这些区域有：Amazon GameLift Servers FleetIQ 指标`PercentUtilizedGameServers`是衡量当前正在使用的托管容量的指标。大多数游戏希望维护未使用的游戏服务器的缓冲区，以便新玩家能够快速进入游戏。您可以创建一个保持特定缓冲区大小的扩展策略，随着玩家需求的波动添加或删除实例。有关更多信息，请参阅 Amazon A EC2 uto Scaling 用户指南中的目标跟踪扩展策略。
- 在开发和生产环境中使用不同的 Amazon 帐户。跨账户将您的开发配置与生产配置分开，可以减少影响现场玩家的配置错误风险。
- 为使用中的游戏服务器组启用游戏会话保护。为了保护玩家，请打开游戏会话保护，防止活动游戏会话因扩展或平衡活动而提前终止。
- 在将游戏与集成 EC2 之前，请先对其进行测试 Amazon GameLift Servers FleetIQ。 我们建议您先启动并运行游戏 EC2 并微调配置。然后，您可以使用相同的启动模板和 AMI 创建游戏服务器组。

  如果您使用的是 Kubernetes，我们建议您先将标准 EC2 实例添加到 Kubernetes 集群，然后使用您为 Kubernetes 集群中的工作节点创建的启动模板创建游戏服务器组。如果您使用的是 EKS，请单独创建 EKS 集群和游戏服务器组。对于游戏服务器组，请将 EKS 优化的 AMI 与合适的用户数据以及用于 EKS 集成的启动模板配置结合使用。在 Amazon EKS 优化 Linux AMI 指南中可查看有关 EKS 工作线程节点和 EKS 优化 AMI 的更多信息。

- 使用游戏服务器组平衡策略 **ON_DEMAND_ONLY** 获得可靠的游戏服务器可用性。此平衡策略生效后，将不使用竞价型实例。这是一个有用的工具，可以确保服务器在您最需要的时候（例如在特征发布或其他特殊活动期间）的可用性。您可以根据需要将游戏服务器组从竞价型策略切换到按需型策略。

另请查看以下 Amazon 最佳实践：

- Amazon 的最佳实践 EC2
- Amazon A EC2 uto Scaling 的最佳实践

Amazon GameLift Servers FleetIQ 功能

- 优化了竞价平衡。Amazon GameLift Servers FleetIQ 定期评估您的实例类型并替换因游戏会话中断的可能性更高而被认为不可行的竞价型实例。当您的 EC2 Auto Scaling 组停用旧实例并启动新实例时，该组会不断刷新当前适用于游戏托管的实例类型。
- 最佳玩家路线。Amazon GameLift Servers FleetIQ APIs 将新的游戏会话定向到弹性最强的竞价型实例，在这些实例中，它们不太可能被中断。此外，游戏会话被打包到更少的实例上，这提高了 EC2 Auto Scaling 组缩减不需要的资源和降低托管成本的能力。
- 根据玩家使用情况自动扩展。Amazon GameLift Servers FleetIQ 以 Amazon CloudWatch 指标形式发布游戏服务器使用率数据。您可以使用这些指标来自动扩展您的可用托管资源，以跟踪实际玩家需求并降低托管成本。
- 直接管理 Amazon EC2 实例。保持对您中的 EC2 实例和 EC2 Auto Scaling 组的完全控制 Amazon Web Services 账户。这意味着您可以设置实例启动模板、维护 EC2 Auto Scaling 组配置以及与其他 Amazon 服务集成。作为其现货平衡活动的一部分，Amazon GameLift Servers FleetIQ 定期更新 Auto Scaling 的某些 EC2 Auto Scaling 组属性。您可以暂时覆盖这些设置或暂停 Amazon GameLift Servers FleetIQ 根据需要进行活动。
- 支持多种游戏服务器可执行文件格式。Amazon GameLift Servers FleetIQ 支持目前在亚马逊上运行的所有格式，包括 Windows EC2、Linux、容器和 Kubernetes。 EC2 FAQs有关支持的操作系统和运行时的列表，请参阅 Amazon。
- 多种类型的托管资源。With Amazon GameLift Servers FleetIQ，您可以访问用于托管游戏服务器的大量实例类型。（可用性因 Amazon 地区而异。）这意味着，您可以将您的游戏服务器与适当的 CPU、内存、存储和网络容量组合进行配对，以便为您的玩家提供最佳的游戏体验。
- 覆盖全球。Amazon GameLift Servers FleetIQ 已在 15 个地区上市，包括中国。这种覆盖范围意味着玩家无论身处何处都能够以最小的游戏延迟访问您提供的游戏服务器。有关区域的完整列表，请参阅 Amazon GameLift Servers中的端点和配额Amazon Web Services 一般参考。

的定价 Amazon GameLift ServersFleetIQ

Amazon GameLift Servers 按使用时长对实例收费，按传输的数据量对带宽收费。有关费用和价格的完整清单 Amazon GameLift Servers，请参阅 Amazon GameLift Servers 定价。

有关计算托管游戏或与之配对的成本的信息 Amazon GameLift Servers，参见生成 Amazon GameLift Servers 定价估算，其中描述了如何使用Amazon 定价计算器。

Amazon GameLift Servers FleetIQ 设置

本节中的主题有助于完成设置任务，包括如何设置您的 Amazon 账户以在 Amazon 上使用 Amazon GameLift Servers FleetIQ 服务。

主题

- Amazon GameLift Servers FleetIQ 支持的软件
- 设置您的 Amazon 账号用于 Amazon GameLift Servers FleetIQ

Amazon GameLift Servers FleetIQ 支持的软件

Amazon GameLift Servers FleetIQ 用于部署 64 位多人游戏服务器、客户端和游戏服务，以便在 Amazon EC2 上托管。此解决方案支持以下环境：

游戏服务器的操作系统

您可以使用 … Amazon GameLift Servers FleetIQ 游戏服务器可在所支持的任何操作系统上运行 EC2。其中包括 Amazon Linux、Ubuntu、Windows Server、Red Hat Enterprise Linux、SUSE Linux Enterprise Server、Fedora、Debian、CentOS、Gentoo Linux、Oracle Linux 和 FreeBSD。在 Amazon EC2 功能中查看当前 EC2 功能和支持。

使用容器

如果您的游戏服务器使用容器，Amazon GameLift Servers FleetIQ 支持与 Kubernetes、亚马逊弹性容器服务 (Amazon ECS) 和亚马逊 Elastic Kubernetes 服务 (EKS) 集成。请在 Amazon上的容器中查看更多信息。

游戏开发环境

游戏客户端和服务器需要一些集成才能与之通信 Amazon GameLift Servers FleetIQ 服务。游戏对 Amazon SDK 进行 API 调用。下载 S Amazon DK 或查看 Amazon GameLift Servers API 参考文档。

支持 Amazon 的 SDK Amazon GameLift Servers 有以下语言版本。有关开发环境支持的信息，请参阅每种语言的文档。

- C++（软件开发工具包文档）(Amazon GameLift Servers)
- Java（软件开发工具包文档）(Amazon GameLift Servers)

- .NET（软件开发工具包文档）(Amazon GameLift Servers)
- Go (SDK 文档) (Amazon GameLift Servers)
- Python（软件开发工具包文档）(Amazon GameLift Servers)
- Ruby（软件开发工具包文档）(Amazon GameLift Servers)
- PHP（软件开发工具包文档）(Amazon GameLift Servers)
- JavaScript/Node.js（SDK 文档）(Amazon GameLift Servers)

设置您的 Amazon 账号用于 Amazon GameLift Servers FleetIQ

要将 Amazon GameLift Servers FleetIQ 对于亚马逊 EC2、Auto Scaling 和其他 Amazon 服务，您必须设置 Amazon Web Services 账户 具有所需访问权限的。完成以下任务：

- 如果你还没有 Amazon 账号可以使用 Amazon GameLift Servers FleetIQ，创建一个新的。请参阅创建一个 Amazon Web Services 账户。
- 设置 Amazon GameLift Servers FleetIQ用户和用户组的特定权限。请参阅 管理用户权限 Amazon GameLift Servers FleetIQ。
- 创建 IAM 角色以允许 Amazon GameLift Servers 以及您的 Amazon 互动 EC2 资源。请参阅 为跨服务交互创建 IAM 角色。

创建一个 Amazon Web Services 账户

创建并设置 Amazon Web Services 账户 要与一起使用 Amazon GameLift Servers FleetIQ。 创建. 不收取任何费用 Amazon Web Services 账户。

主题

- 注册获取 Amazon Web Services 账户
- 保护 IAM 用户

注册获取 Amazon Web Services 账户

如果您没有 Amazon Web Services 账户，请完成以下步骤来创建一个。

报名参加 Amazon Web Services 账户

1. 打开https://portal.aws.amazon.com/billing/注册。

2. 按照屏幕上的说明操作。

   在注册时，将接到电话，要求使用电话键盘输入一个验证码。

   当您注册时 Amazon Web Services 账户，就会创建Amazon Web Services 账户根用户一个。根用户有权访问该账户中的所有 Amazon Web Services 服务 和资源。作为最佳安全实践，请为用户分配管理访问权限，并且只使用根用户来执行需要根用户访问权限的任务。

Amazon 注册过程完成后会向您发送一封确认电子邮件。您可以随时前往 https://aws.amazon.com/并选择 “我的账户”，查看您当前的账户活动并管理您的账户。

保护 IAM 用户

注册后 Amazon Web Services 账户，开启多重身份验证 (MFA)，保护您的管理用户。有关说明，请参阅《IAM 用户指南》中的 为 IAM 用户启用虚拟 MFA 设备（控制台）。

要允许其他用户访问您的 Amazon Web Services 账户 资源，请创建 IAM 用户。为了保护您的 IAM 用户，请启用 MFA 并仅向 IAM 用户授予执行任务所需的权限。

有关创建和保护 IAM 用户的更多信息，请参阅《IAM 用户指南》中的以下主题：

- 在你的 IAM 用户中创建 Amazon Web Services 账户
- 适用于 Amazon 资源的访问权限管理
- 基于 IAM 身份的策略示例

管理用户权限 Amazon GameLift Servers FleetIQ

创建其他用户或扩展 Amazon GameLift Servers FleetIQ 根据需要向现有用户提供访问权限。与之合作的用户 Amazon GameLift Servers FleetIQ 游戏服务器组以及相关的 Amazon EC2 和 Auto Scaling 服务必须具有访问这些服务的权限。

作为最佳实操（IAM 中的安全最佳实操），请为所有用户应用最低权限权限。您可以为各个用户或用户组设置权限，并按服务、操作或资源限制用户访问权限。

根据您管理 Amazon 账户中用户的方式，按照以下说明设置用户权限。如果您使用 IAM 用户，则作为最佳实操，请始终向角色或用户组授予权限，而不是向个人用户授予权限。

- 用户的权限语法

- 与一起使用的其他权限语法 Amazon CloudFormation

要提供访问权限，请为您的用户、组或角色添加权限：

- 通过身份提供商在 IAM 中托管的用户：

  创建适用于身份联合验证的角色。按照《IAM 用户指南》中针对第三方身份提供商创建角色（联合身份验证）的说明进行操作。

- IAM 用户：
  - 创建您的用户可以担任的角色。按照《IAM 用户指南》中为 IAM 用户创建角色的说明进行操作。
  - （不推荐使用）将策略直接附加到用户或将用户添加到用户组。按照《IAM 用户指南》中向用户添加权限（控制台）中的说明进行操作。

参考：Amazon GameLift Servers FleetIQ_策略

以下是示例 Amazon GameLift Servers FleetIQ_policy 供你参考：

```
{
  "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": "*"
    }
  ]
```

```
}
```

的额外权限 Amazon CloudFormation

如果您使用 Amazon CloudFormation管理游戏托管资源，请将 Amazon CloudFormation 权限添加到策略语法中。

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

为用户设置编程式访问权限

如果用户想在 Amazon 外部进行交互，则需要编程访问权限 Amazon Web Services Management Console。 Amazon APIs 和 Amazon Command Line Interface 需要访问密钥。可能的话，创建临时凭证，该凭证由一个访问密钥 ID、一个秘密访问密钥和一个指示凭证何时到期的安全令牌组成。

要向用户授予编程式访问权限，请选择以下选项之一。

哪个用户需要编程式访问权限？	目的	方式
IAM	使用短期证书签署对 Amazon CLI 或的编程请求 Amazon APIs（直接或使用 Amazon SDKs）。	按照 IAM 用户指南中的将临时证书与 Amazon 资源配合使用中的说明进行操作。
IAM	（不推荐使用） 使用长期证书签署对 Amazon CLI 或的编程请求 Amazon APIs（直接或使用 Amazon SDKs）。	按照《IAM 用户指南》中管理 IAM 用户的访问密钥中的说明进行操作。

如果您使用访问密钥，请参阅管理 Amazon 访问密钥的最佳实践。

为跨服务交互创建 IAM 角色

为了 Amazon GameLift Servers FleetIQ 要使用您的 Amazon EC2 实例和 Auto Scaling 群组，您必须允许服务相互交互。这是通过在您的 Amazon 账户中创建 IAM 角色并分配一组有限权限来完成的。每个角色还指定能够担任此角色的服务。

设置以下角色：

- 为创建角色 Amazon GameLift Servers FleetIQ更新您的 Amazon EC2 资源。
- 为 Amazon 创建角色 EC2可与之沟通的资源 Amazon GameLift Servers FleetIQ.

为创建角色 Amazon GameLift Servers FleetIQ

这个角色允许 Amazon GameLift Servers FleetIQ 访问和修改您的 Amazon EC2 实例、Auto Scaling 组和生命周期挂钩，这是其竞价平衡和自动扩展活动的一部分。

使用 IAM 控制台或 Amazon CLI 为创建角色 Amazon GameLift Servers FleetIQ 并附加具有必要权限的托管策略。有关 IAM 角色和托管策略的更多信息，请参阅为 Amazon 服务创建角色和Amazon 托管策略。

Console

以下步骤介绍如何使用托管策略创建服务角色 Amazon GameLift Servers 使用 Amazon Web Services Management Console.

1. 登录 IAM 控制台，然后选择角色: 创建角色。
2. 对于Select type of trusted entity（选择受信任实体的类型），选择 Amazon service（服务）。
3. 对于选择用例，请GameLift从服务列表中进行选择。在 “选择您的用例” 下，相应的 Amazon GameLift Servers 自动选择用例。要继续，请选择下一步：权限。
4. 列表附加的权限策略应包含一个策略:GameLiftGameServerGroupPolicy . 如果未显示此策略，请检查筛选条件或使用搜索功能将其添加到角色中。您可以查看策略的语法（选择 ▸ 图标以展开），但不能更改语法。创建角色后，您可以更新角色并附加其他策略以添加或删除权限。

   对于设置权限边界，保留默认设置（创建没有权限边界的角色）。这是一个不需要的高级设置。要继续，请选择下一步：标签。

5. 添加标签是用于资源管理的可选设置。例如，您可能希望向此角色添加标签，以便按角色跟踪特定于项目的资源使用情况。要查看有关为 IAM 角色和其他用途进行标记的更多信息，请访问了解更多链接。要继续，请选择下一步：审核。
6. 在审核页上，根据需要进行以下更改：
   - 输入角色名称并（可选）更新说明。
   - 请验证以下内容：
     - 可信实体设置为 “Amazon 服务：gamelift.amazonaws.com”。创建角色后，必须更新此值。
     - 政策包括 GameLiftGameServerGroupPolicy。

   要完成任务，请选择创建角色。
7. 创建新角色后，您必须手动更新角色的信任关系。转到角色页面并选择新角色名称以打开其摘要页面。打开信任关系选项卡，然后选择编辑信任关系。在策略文档中，更新 `Service` 属性以包含 `autoscaling.amazonaws.com`。修订后的 `Service` 属性应如下所示：

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

   要保存更改，请选择更新信任策略。

该角色已准备就绪。记下角色的 ARN 值，该值显示在角色摘要页面的顶部。设置时您将需要这些信息 Amazon GameLift Servers FleetIQ 游戏服务器组。

Amazon CLI

以下步骤介绍如何使用托管策略创建服务角色 Amazon GameLift Servers 使用 C Amazon LI。

1. 使用以下 JSON 语法创建信任策略文件（示例：`FleetIQtrustpolicyGameLift.json`）。

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

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

2. 使用 iam create-role 创建一个新的 IAM 角色，并将其与您刚创建的信任策略 JSON 文件关联。

   Windows:

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

   Linux：

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

   请求成功时，响应将包括新创建的角色的属性。记下 ARN 值。设置时您将需要这些信息 Amazon GameLift Servers FleetIQ 游戏服务器组。

3. 使用 iam 附加托管权限策略 “GameLiftGameServerGroupPolicy”。 attach-role-policy

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

   要验证权限策略是否已附加，请使用新角色 list-attached-role-policies的名称调用 iam。

该角色已准备就绪。您可以通过调用 gamelift create-game-server-group 来验证 IAM 角色的配置是否正确，并将`role-arn`属性设置为新角色的 ARN 值。当`GameServerGroup`进入活动状态时，这表明 Amazon GameLift Servers FleetIQ 能够按预期修改您账户中的 Amazon EC2 和 Auto Scaling 资源。

为 Amazon 创建角色 EC2

此角色使您的 Amazon EC2 资源能够与之通信 Amazon GameLift Servers FleetIQ。 例如，在 Amazon EC2 实例上运行的游戏服务器需要能够报告运行状况。创建 Amazon EC2 启动模板时，将此角色包含在 IAM 实例配置文件中 Amazon GameLift Servers FleetIQ 游戏服务器组。

使用 Amazon CLI 为 Amazon 创建角色 EC2，附加具有必要权限的自定义策略，并将该角色附加到实例配置文件。有关更多信息，请参阅为 Amazon 服务创建角色。

Amazon CLI

以下步骤描述了如何使用自定义创建服务角色 Amazon GameLift Servers Amazon EC2 使用权限 Amazon CLI。

1. 使用以下 JSON 语法创建信任策略文件（示例：`FleetIQtrustpolicyEC2.json`）。

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

2. 使用 iam create-role 创建一个新的 IAM 角色，并将其与您刚创建的信任策略 JSON 文件关联。

   Windows:

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

   Linux：

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

请求成功时，响应将包括新创建的角色的属性。记下 ARN 值。在设置 Amazon EC2 启动模板时，您将需要这些信息。

3. 使用以下 JSON 语法创建权限策略文件（示例：`FleetIQpermissionsEC2.json`）。

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

4. 使用 iam put-role-policy 将您刚刚创建的权限策略 JSON 文件附加到新角色。

Windows:

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

Linux：

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

要验证权限策略是否已附加，请使用新角色 list-role-policies的名称调用 iam。

5. 使用 iam 创建带有新角色 create-instance-profile的实例配置文件以用于 Amazon EC2。有关更多信息，请参阅管理实例配置文件。

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

当请求成功时，响应将包括新创建的实例配置文件的属性。

6. 使用 add-role-to-instanceiam-profil e 将角色附加到实例配置文件。

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

该角色和个人资料现已准备就绪，可以与 Amazon EC2 启动模板一起使用。

正在为游戏做准备 Amazon GameLift Servers FleetIQ

本节介绍如何通过以下方式实现在 Amazon 上托管游戏 EC2 的设计 Amazon GameLift Servers FleetIQ 要启动并运行多人游戏，您需要执行以下操作：

- 调整您的游戏服务器以与之通信 Amazon GameLift Servers FleetIQ.
- 创建一个 FleetIQ 游戏服务器组来部署您的游戏服务器。
- 向游戏客户端服务添加功能以请求可用的游戏服务器。

本节中的主题介绍如何完成这项工作的详细信息。要开始使用，请参阅集成计划，其中提供了详细的 step-by-step指南。

主题

- Amazon GameLift Servers FleetIQ 集成步骤
- 管理 Amazon GameLift Servers FleetIQ 游戏服务器组
- 集成 Amazon GameLift Servers FleetIQ 进入游戏服务器
- 集成 Amazon GameLift Servers FleetIQ 进入游戏客户端

Amazon GameLift Servers FleetIQ 集成步骤

该集成计划概述了在亚马逊 EC2 实例上启动和运行多人游戏的关键步骤 Amazon GameLift Servers FleetIQ。 如果你正在寻找 Amazon GameLift Servers 托管托管服务，可为您自动执行更多游戏托管流程，请参阅 Amazon GameLift Servers 开发者指南。

要开始使用 Amazon GameLift Servers FleetIQ，你需要有一台能在本地或 Amazon EC2 环境中运行的游戏服务器。游戏服务器可以是管理一个或多个游戏会话、生成子进程或在容器内运行的单个进程。

1. 获取Amazon 账户并使用设置用户 Amazon GameLift Servers FleetIQ 访问。

   创建一个新的账户 Amazon Web Services 账户 或选择一个现有账户来使用 Amazon GameLift Servers FleetIQ。 为用户设置管理您的游戏中使用的亚马逊 EC2、Auto Scaling 和其他 Amazon 资源的权限。有关详细说明，请参阅设置您的 Amazon 账号用于 Amazon GameLift Servers FleetIQ。
2. 创建 IAM 角色。

   创建允许的角色 Amazon GameLift Servers FleetIQ EC2、Amazon 和 Auto Scaling 资源可以相互通信。有关更多信息，请参阅为跨服务交互创建 IAM 角色。

3. 使用 Amazon SDK 和 Amazon CLI 获取 Amazon GameLift Servers FleetIQ 功能。
   - 下载最新版本的 Amazon 软件开发工具包。
   - 查看 Amazon GameLift Servers API 参考文档。
4. 准备好游戏服务器以供使用 Amazon GameLift Servers FleetIQ.

   将 Amazon SDK 添加到您的游戏服务器项目并添加要保留的代码 Amazon GameLift Servers FleetIQ 更新了游戏服务器的当前状态和使用情况。有关其他指导和示例，请参阅the section called “集成游戏服务器”。Amazon GameLift Servers FleetIQ 使用此信息为您的配对系统提供可行、未占用的游戏服务器列表，还可以避免在平衡期间终止当前托管玩家的实例。
5. 使用您的游戏服务器创建 EC2 亚马逊系统映像 (AMI)。

   使用您的游戏服务器软件以及任何其他运行时资产或配置设置创建 AMI。如需帮助，请参阅亚马逊 EC2 用户指南中的亚马逊系统映像 (AMI)。
6. 创建 Amazon EC2 启动模板。

   构建 Amazon EC2 启动模板，该模板使用您的自定义 AMI，并为您的托管资源定义网络和安全设置。启动模板必须引用您创建的实例配置文件（参见步骤 2），其权限允许您的游戏服务器与之通信 Amazon GameLift Servers FleetIQ。 您无需在启动模板中包含实例类型，因为这将在稍后完成。如需帮助，请参阅 Amazon EC2 用户指南中的创建启动模板。

   > **Note**
   > 在使用启动模板之前 Amazon GameLift Servers FleetIQ，我们强烈建议您先设置一个 Auto Scaling 组，以验证模板配置和 AMI 的部署是否正确。
7. 设置 Amazon GameLift Servers FleetIQ 托管资源。

   在要部署游戏服务器的每个区域中，通过调用 CreateGameServerGroup() 创建游戏服务器组。传入启动模板（包含您的自定义 AMI 以及网络和安全设置）、IAM 角色以及游戏可以运行的实例类型列表。此操作会在您的 Amazon 账户中设置一个 Auto Scaling 群组 Amazon GameLift Servers FleetIQ 可以修改。有关其他指导和示例，请参阅管理 Amazon GameLift Servers FleetIQ 游戏服务器组。
8. 整合 Amazon GameLift Servers FleetIQ 进入你的游戏客户端。

   将 Amazon SDK 添加到您的游戏客户端、媒人或其他分配游戏服务器容量的后端组件中。根据您的游戏类型，您的媒人可能会调用 ListGameServers() 或 ClaimGameServer() 来获取服务器容量并保

留可用的游戏服务器。有关其他指导和示例，请参阅集成 Amazon GameLift Servers FleetIQ 进入游戏客户端。

9. 扩展您的自动扩缩组。

   在您的自动扩缩组中预配置实例时，它们会启动您的游戏服务器。然后，每台游戏服务器都注册到 Amazon GameLift Servers FleetIQ 作为可用容量，供您的媒人稍后列出或申领。

10. 测试游戏。

    调用您的对战构建器并调用 `ClaimGameServer` 以请求服务器容量。将生成的 IP 和端口传回游戏客户端，以便他们可以连接到游戏服务器。

管理 Amazon GameLift Servers FleetIQ 游戏服务器组

本主题描述了设置所需的任务 Amazon GameLift Servers FleetIQ 游戏服务器组。创建游戏服务器组会触发一个 EC2 Auto Scaling 组的创建，该组包含所有必要的配置设置以及要管理的配置 Amazon GameLift Servers FleetIQ 游戏托管优化。

在创建游戏服务器组之前，您必须至少准备好以下资源：

- Amazon EC2 启动模板，用于指定如何使用您的游戏服务器版本启动 Amazon EC2 实例。有关更多信息，请参阅 Amazon EC2 用户指南中的从启动模板启动实例。
- 一个 IAM 角色，可将有限访问权限扩展到您的 Amazon 账户，以允许 Amazon GameLift Servers FleetIQ 创建 Auto Scaling 组并与之交互。有关更多信息，请参阅 为跨服务交互创建 IAM 角色。

创建游戏服务器组

要创建游戏服务器组，请调用 CreateGameServerGroup()。此操作会同时创建 Amazon GameLift Servers FleetIQ 游戏服务器组和相应的 Auto Scaling 组。创建游戏服务器组时，您需要为以下项提供特定于游戏的设置 Amazon GameLift Servers FleetIQ，包括平衡策略和实例类型定义。您还为自动扩缩组提供初始属性设置。

以下示例触发创建 `GameServerGroup`，其指定 c4.large 和 c5.large 实例类型并将该组限制为仅限竞价型实例，以及一个自动扩缩组，该组使用指定的启动模板部署实例，并使用目标跟踪自动扩展策略在最小和最大设置范围内管理组容量。短暂的预配置时段后，将创建一个 `AutoScalingGroup` 资源，并且 `GameServerGroup` 进入 ACTIVE 状态。

```
Amazon 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}}'
```

更新游戏服务器组

您可以更新游戏服务器组属性，这些属性会影响游戏服务器组的方式 Amazon GameLift Servers FleetIQ 管理游戏服务器的托管，包括资源类型优化。要更新这些属性，请调用 UpdateGameServerGroup()。对游戏服务器组的更改生效后，Amazon GameLift Servers FleetIQ 可能会覆盖 Auto Scaling 组中的某些属性。

对于所有其他自动扩缩组属性，例如`MinSize`、`MaxSize`、和 `LaunchTemplate`，您可以直接在自动扩缩组上修改这些属性。

在下面的示例中，更新实例类型定义以切换到 c4.xlarge 和 c5.xlarge 实例类型。

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

跟踪游戏服务器组实例

在创建实例并将其部署到游戏服务器组和 Auto Scaling 组之后，您可以通过调用 DescribeGameServerInstances() 来跟踪游戏服务器实例的状态。您可以使用此操作来跟踪实例状态。有关游戏服务器组状态的更多信息，请参阅游戏服务器组的生命周期。

您也可以使用 Amazon GameLift Servers 控制台，位于游戏服务器组下，用于监视游戏服务器组的状态。

集成 Amazon GameLift Servers FleetIQ 进入游戏服务器

本主题介绍准备要与之通信的游戏服务器项目所需的任务 Amazon GameLift Servers FleetIQ。 有关其他指导Amazon GameLift Servers FleetIQ 最佳实践，请参阅。

注册游戏服务器

当游戏服务器进程启动并准备好托管实时游戏时，它必须向注册 Amazon GameLift Servers FleetIQ 通过调用 RegisterGameServer()。注册允许 Amazon GameLift Servers FleetIQ 在配对系统或其他客户端服务请求服务器容量信息或申领游戏服务器时对其做出响应。注册时，游戏服务器可以提供 Amazon GameLift Servers FleetIQ 包含相关的游戏服务器数据和连接信息，包括用于入站客户端连接的端口和 IP 地址。

```
Amazon 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\"}"
```

更新游戏服务器状态

注册游戏服务器后，它应定期报告运行状况和利用率状态，以保持服务器容量状态同步 Amazon GameLift Servers FleetIQ。 通过调用 UpdateGameServer() 报告运行状况和利用率状态。在下面的示例中，游戏服务器报告正常运行，并且未由玩家或游戏占用。

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

运行状况

如果您的游戏服务器有跟踪生命值状态的机制，则可以使用此机制来触发游戏服务器生命值更新 Amazon GameLift Servers FleetIQ.

利用率状态

报告游戏服务器利用率状态保持 Amazon GameLift Servers FleetIQ 告知目前哪些游戏服务器是理想的，可用于新游戏会话。您的游戏服务器必须具有触发利用率状态更新的机制 Amazon GameLift Servers FleetIQ。 例如，当玩家连接到游戏服务器或游戏会话开始时，你可能会触发更新。

开始游戏会话时，客户端或配对服务会话会声明可用的游戏服务器 ClaimGameServer（通过调用 ()），提示玩家连接到游戏服务器，并触发游戏服务器开始游戏。集成 Amazon GameLift Servers FleetIQ 进入游戏客户端 中介绍了此过程。游戏服务器“声明”的有效期为 60 秒，并且游戏服务器必须能够在此窗口内更新利用率状态。如果未更新利用率状态，Amazon GameLift Servers FleetIQ 移除声明，假设游戏服务器可用，并可能为其他客户端索赔请求保留游戏服务器。

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

注销游戏服务器

游戏结束后，游戏服务器必须从中取消注册 Amazon GameLift Servers FleetIQ 使用 DeregisterGameServer()。

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

集成 Amazon GameLift Servers FleetIQ 进入游戏客户端

本主题描述了准备与之通信的游戏客户端或配对服务所需的任务 Amazon GameLift Servers FleetIQ 以获取用于托管游戏会话的游戏服务器。

创建一种方法，允许游戏客户端或对战构建器为玩家请求游戏服务器资源。为此，您可以在几种方法中选择：

- 有 Amazon GameLift Servers FleetIQ 选择可用的游戏服务器。此选项利用了 Amazon GameLift Servers FleetIQ 针对使用低成本竞价型实例和自动扩展进行了优化。
- 请求所有可用的游戏服务器并选择一个以使用（通常称为“列出并挑选”）。

主题

- 让 Amazon GameLift Servers FleetIQ 选择游戏服务器
- 选择您自己的游戏服务器

让 Amazon GameLift Servers FleetIQ 选择游戏服务器

要拥有 Amazon GameLift Servers FleetIQ 选择可用的游戏服务器，在不指定游戏服务器 ID 的情况下调用 ClaimGameServer()。在这种情况下，Amazon GameLift Servers FleetIQ 确实会运用其逻辑，在可进行游戏托管并针对自动扩展进行优化的实例上找到游戏服务器。

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

在回应索赔请求时，Amazon GameLift Servers FleetIQ 标识`GameServer`资源、连接信息和游戏数据，客户端可以使用这些资源连接到游戏服务器。游戏服务器的认领状态设置为 CLAIMED 60 秒。您的游戏服务器或客户端服务需要更新游戏服务器的状态 Amazon GameLift Servers FleetIQ 在玩家连接或游戏开始之后。这样可以确保 Amazon GameLift Servers FleetIQ 不提供此游戏服务器来响应后续的游戏服务器容量请求。通过调用 UpdateGameServer() 更新游戏服务器状态。

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

选择您自己的游戏服务器

使用 “列出并选择” 方法，您的游戏客户端或媒人通过调用 ListGameServers() 来请求可用游戏服务器的列表。在选择游戏服务器时，您可能希望使用游戏服务器数据来提供玩家或对战构建器可以使用的其他信息。要控制结果的返回方式，您可以请求分页结果并按注册日期对游戏服务器进行排序。以下请求返回指定游戏服务器组中的 20 个处于活动状态且可用的游戏服务器，按注册时间排序，最新的游戏服务器排在最前面。

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

根据可用游戏服务器的列表，客户端或配对服务选择一个游戏服务器，然后使用特定的游戏服务器 ID 调用 ClaimGameServer() 来声明该服务器。在这种情况下，Amazon GameLift Servers FleetIQ 不执行其任何实例类型优化逻辑，如中所述Amazon GameLift Servers FleetIQ 逻辑。

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

监控 Amazon GameLift Servers FleetIQ 与亚马逊合作 CloudWatch

使用 Amazon CloudWatch 指标来扩展您的实例容量、构建操作控制面板和触发警报。Amazon GameLift Servers FleetIQ 作为独立解决方案，会向您的 Amazon 账户发送一组 Amazon CloudWatch 指标。另请参阅 Amazon A uto Scaling 用户指南 CloudWatch中的使用亚马逊监控您的 EC2 自动缩放组和实例。

这些区域有：FleetIQ 此处列出了指标。请查看完整的亚马逊 CloudWatch 指标信息 Amazon GameLift Servers 在 Amazon GameLift Servers 指标。

指标	描述
`AvailableGameServers`	可用于运行游戏执行但当前未被玩游戏占用的游戏服务器数量。此数字包括已认领但仍处于 AVAILABLE（可用）状态的游戏服务器。 单位：计数 亚马逊的相关 CloudWatch 统计数据：总和 尺寸： GameServerGroup
`UtilizedGameServers`	当前被游戏占用的游戏服务器数量。此数字包括处于 UTILIZED 状态的游戏服务器。 单位：计数 亚马逊的相关 CloudWatch 统计数据：总和 尺寸： GameServerGroup
`DrainingAvailableGameServers`	当前不支持玩游戏且计划终止的实例上的游戏服务器数量。这些游戏服务器属于为响应新的认领请求而认领的最低优先级。 单位：计数 亚马逊的相关 CloudWatch 统计数据：总和

指标	描述
	尺寸： GameServerGroup
`DrainingUtilizedGameServers`	当前支持玩游戏且计划终止的实例上的游戏服务器数量。 单位：计数 亚马逊的相关 CloudWatch 统计数据：总和 尺寸： GameServerGroup
`PercentUtilizedGameServers`	当前支持游戏执行的游戏服务器所占的部分。此指标表示当前正在使用的游戏服务器容量。它可用于促使制定相应的自动扩缩策略，以便可以动态添加和删除实例来与玩家需求匹配。 单位：百分比 Amazon 的相关 CloudWatch 统计数据：平均值、最小值、最大值 尺寸： GameServerGroup
`GameServerInterruptions`	由于竞价型实例可用性有限而中断的竞价型实例上的游戏服务器数量。 单位：计数 亚马逊的相关 CloudWatch 统计数据：总和 尺寸： GameServerGroup， InstanceType
`InstanceInterruptions`	由于可用性有限而中断的竞价型实例数量。 单位：计数 亚马逊的相关 CloudWatch 统计数据：总和 尺寸： GameServerGroup， InstanceType

安全性与 Amazon GameLift Servers FleetIQ

如果你正在使用 Amazon GameLift Servers FleetIQ 作为亚马逊的一项独立功能 EC2，请参阅亚马逊 EC2 用户指南 EC2中的亚马逊安全。

云安全 Amazon 是重中之重。为了满足对安全性最敏感的组织的需求，我们打造了具有超高安全性的数据中心和网络架构。作为 Amazon 的客户，您也可以从这些数据中心和网络架构受益。

安全是双方共同承担 Amazon 的责任。有关在使用分担责任模型时如何应用分担责任模型的信息 Amazon GameLift Servers FleetIQ，请参阅中的 “安全” Amazon GameLift Servers.

Amazon GameLift Servers FleetIQ 参考指南

本节包含用于的参考文档 Amazon GameLift Servers FleetIQ.

主题

- Amazon GameLift Servers FleetIQ 服务 API 参考 (Amazon SDK)
- Amazon GameLift Servers FleetIQ 发行说明和 SDK 版本
- Amazon GameLift Servers 开发者资源

Amazon GameLift Servers FleetIQ 服务 API 参考 (Amazon SDK)

本主题提供了基于任务的 API 操作列表 Amazon GameLift Servers FleetIQ。 的 Amazon GameLift Servers FleetIQ 服务 API 打包到`aws.gamelift`命名空间中的 Amazon SDK 中。下载 S Amazon DK 或查看 Amazon GameLift Servers API 参考文档。

Amazon GameLift Servers FleetIQ 优化了低成本竞价型实例的使用，以便在 Amazon EC2 上托管基于云的游戏。请参阅Amazon GameLift Servers 开发者指南，了解有关其他内容的更多信息 Amazon GameLift Servers 托管选项。

主题

- Amazon GameLift Servers FleetIQ API 操作
- 可用编程语言

Amazon GameLift Servers FleetIQ API 操作

以下操作允许您管理自己的 Amazon GameLift Servers FleetIQ 资源，包括游戏服务器组和游戏服务器，以及 Amazon EC2 和 Auto Scaling 群组。

管理游戏服务器组

使用这些操作来管理您的游戏服务器部署 FleetIQ 优化。游戏服务器组控制您的游戏服务器进程如何在 Amazon EC2 实例、设置和 Auto Scaling 组上启动，并定义如何应用 FleetIQ 优化。

- CreateGameServerGroup— 创建新的游戏服务器组和相应的 Auto Scaling 组，然后开始启动用于托管游戏服务器的实例。CLI 命令：create-game-server-group

- ListGameServerGroups— 获取一个中所有游戏服务器组的列表 Amazon GameLift Servers region。CLI 命令：list-game-server-groups
- DescribeGameServerGroup— 检索游戏服务器组的元数据。CLI 命令：describe-game-server-group
- UpdateGameServerGroup— 更改游戏服务器组元数据。CLI 命令：update-game-server-group
- DeleteGameServerGroup— 永久移除游戏服务器组并终止 FleetIQ 关联托管资源的活动。CLI 命令：delete-game-server-group
- ResumeGameServerGroup— 恢复已暂停 FleetIQ 游戏服务器组的活动。CLI 命令：resume-game-server-group
- SuspendGameServerGroup— 暂时停止 FleetIQ 游戏服务器组的活动。CLI 命令：suspend-game-server-group

管理游戏服务器

使用这些操作来管理您的游戏服务器部署 FleetIQ 优化。游戏服务器组控制您的游戏服务器进程如何在 Amazon EC2 实例、设置和 Auto Scaling 组上启动，并定义如何应用 FleetIQ 优化。

- RegisterGameServer— 从新游戏服务器打电话通知 Amazon GameLift Servers FleetIQ 游戏服务器已准备好托管游戏玩法。CLI 命令：register-game-server-group
- ListGameServers— 从游戏客户端服务调用以获取游戏服务器组中当前运行的所有游戏服务器的列表。CLI 命令：list-game-servers
- ClaimGameServer— 从游戏客户端服务中拨打电话，找到并预订游戏服务器以托管新的游戏会话。CLI 命令：claim-game-server
- DescribeGameServer— 检索游戏服务器的元数据。CLI 命令：describe-game-server
- UpdateGameServer— 更改游戏服务器元数据、健康状态或使用状态。CLI 命令：update-game-server
- DeregisterGameServer— 从正在终止的游戏服务器调用以提示 Amazon GameLift Servers FleetIQ 将游戏服务器从游戏服务器组中移除。CLI 命令：deregister-game-server

可用编程语言

支持 Amazon 的 SDK Amazon GameLift Servers 有以下语言版本。有关开发环境支持的信息，请参阅每种语言的文档。

- C++（软件开发工具包文档）(Amazon GameLift Servers)

- Java（软件开发工具包文档）(Amazon GameLift Servers)
- .NET（软件开发工具包文档）(Amazon GameLift Servers)
- Go (SDK 文档) (Amazon GameLift Servers)
- Python（软件开发工具包文档）(Amazon GameLift Servers)
- Ruby（软件开发工具包文档）(Amazon GameLift Servers)
- PHP（软件开发工具包文档）(Amazon GameLift Servers)
- JavaScript/Node.js（SDK 文档）(Amazon GameLift Servers)

Amazon GameLift Servers FleetIQ 发行说明和 SDK 版本

这些区域有：Amazon GameLift Servers 发行说明提供了有关新内容的详细信息 FleetIQ 与该服务相关的功能、更新和修复。本页还包括 Amazon GameLift Servers SDK 版本历史记录。

Amazon GameLift Servers 开发者资源

查看全部 Amazon GameLift Servers 文档和开发者资源，请参阅 Amazon GameLift Servers 文档主页。

Amazon 词汇表

有关最新 Amazon 术语，请参阅《Amazon Web Services 词汇表 参考资料》中的Amazon 词汇表。

本文属于机器翻译版本。若本译文内容与英语原文存在差异，则一律以英文原文为准。