# Table of Contents

Welcome ............................................................................................................................................. 1  
Actions ................................................................................................................................................ 5  
AcceptMatch ................................................................. 7  
  Request Syntax .......................................................... 7  
  Request Parameters .................................................. 7  
  Response Elements ................................................... 8  
  Errors ........................................................................... 8  
  See Also ....................................................................... 9  
CreateAlias ................................................................. 10  
  Request Syntax .......................................................... 10  
  Request Parameters .................................................. 10  
  Response Syntax ....................................................... 11  
  Response Elements ................................................... 11  
  Errors ........................................................................... 11  
  See Also ....................................................................... 12  
CreateBuild ................................................................. 13  
  Request Syntax .......................................................... 13  
  Request Parameters .................................................. 14  
  Response Syntax ....................................................... 14  
  Response Elements ................................................... 15  
  Errors ........................................................................... 15  
  Examples ....................................................................... 16  
  See Also ....................................................................... 18  
CreateFleet ................................................................. 19  
  Request Syntax .......................................................... 20  
  Request Parameters .................................................. 20  
  Response Syntax ....................................................... 24  
  Response Elements ................................................... 24  
  Errors ........................................................................... 24  
  Examples ....................................................................... 25  
  See Also ....................................................................... 29  
CreateGameSession ..................................................... 31  
  Request Syntax .......................................................... 31  
  Request Parameters .................................................. 32  
  Response Syntax ....................................................... 33  
  Response Elements ................................................... 34  
  Errors ........................................................................... 34  
  See Also ....................................................................... 35  
CreateGameSessionQueue .......................................... 37  
  Request Syntax .......................................................... 37  
  Request Parameters .................................................. 37  
  Response Syntax ....................................................... 38  
  Response Elements ................................................... 39  
  Errors ........................................................................... 39  
  Example ......................................................................... 39  
  See Also ....................................................................... 40  
CreateMatchmakingConfiguration ................................ 41  
  Request Syntax .......................................................... 41  
  Request Parameters .................................................. 42  
  Response Syntax ....................................................... 44  
  Response Elements ................................................... 44  
  Errors ........................................................................... 44  
  See Also ....................................................................... 45  
CreateMatchmakingRuleSet ....................................... 46
<table>
<thead>
<tr>
<th>Operation</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>DescribeInstances</td>
<td>126</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>127</td>
</tr>
<tr>
<td>Request Parameters</td>
<td>127</td>
</tr>
<tr>
<td>Response Syntax</td>
<td>128</td>
</tr>
<tr>
<td>Response Elements</td>
<td>128</td>
</tr>
<tr>
<td>Errors</td>
<td>128</td>
</tr>
<tr>
<td>Example</td>
<td>129</td>
</tr>
<tr>
<td>See Also</td>
<td>130</td>
</tr>
<tr>
<td>DescribeMatchmaking</td>
<td>131</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>131</td>
</tr>
<tr>
<td>Request Parameters</td>
<td>131</td>
</tr>
<tr>
<td>Response Syntax</td>
<td>131</td>
</tr>
<tr>
<td>Response Elements</td>
<td>132</td>
</tr>
<tr>
<td>Errors</td>
<td>132</td>
</tr>
<tr>
<td>See Also</td>
<td>133</td>
</tr>
<tr>
<td>DescribeMatchmakingConfigurations</td>
<td>134</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>134</td>
</tr>
<tr>
<td>Request Parameters</td>
<td>134</td>
</tr>
<tr>
<td>Response Syntax</td>
<td>135</td>
</tr>
<tr>
<td>Response Elements</td>
<td>135</td>
</tr>
<tr>
<td>Errors</td>
<td>136</td>
</tr>
<tr>
<td>See Also</td>
<td>136</td>
</tr>
<tr>
<td>DescribeMatchmakingRuleSets</td>
<td>137</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>137</td>
</tr>
<tr>
<td>Request Parameters</td>
<td>137</td>
</tr>
<tr>
<td>Response Syntax</td>
<td>138</td>
</tr>
<tr>
<td>Response Elements</td>
<td>138</td>
</tr>
<tr>
<td>Errors</td>
<td>138</td>
</tr>
<tr>
<td>See Also</td>
<td>138</td>
</tr>
<tr>
<td>DescribePlayerSessions</td>
<td>140</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>140</td>
</tr>
<tr>
<td>Request Parameters</td>
<td>140</td>
</tr>
<tr>
<td>Response Syntax</td>
<td>142</td>
</tr>
<tr>
<td>Response Elements</td>
<td>142</td>
</tr>
<tr>
<td>Errors</td>
<td>142</td>
</tr>
<tr>
<td>See Also</td>
<td>143</td>
</tr>
<tr>
<td>DescribeRuntimeConfiguration</td>
<td>144</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>144</td>
</tr>
<tr>
<td>Request Parameters</td>
<td>144</td>
</tr>
<tr>
<td>Response Syntax</td>
<td>145</td>
</tr>
<tr>
<td>Response Elements</td>
<td>145</td>
</tr>
<tr>
<td>Errors</td>
<td>145</td>
</tr>
<tr>
<td>See Also</td>
<td>146</td>
</tr>
<tr>
<td>DescribeScalingPolicies</td>
<td>147</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>147</td>
</tr>
<tr>
<td>Request Parameters</td>
<td>147</td>
</tr>
<tr>
<td>Response Syntax</td>
<td>148</td>
</tr>
<tr>
<td>Response Elements</td>
<td>149</td>
</tr>
<tr>
<td>Errors</td>
<td>149</td>
</tr>
<tr>
<td>Example</td>
<td>149</td>
</tr>
<tr>
<td>See Also</td>
<td>151</td>
</tr>
<tr>
<td>DescribeVpcPeeringAuthorizations</td>
<td>152</td>
</tr>
<tr>
<td>Response Syntax</td>
<td>152</td>
</tr>
<tr>
<td>Response Elements</td>
<td>152</td>
</tr>
<tr>
<td>Errors</td>
<td>152</td>
</tr>
<tr>
<td>See Also</td>
<td>153</td>
</tr>
<tr>
<td>Function</td>
<td>Page</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>ResolveAlias</td>
<td>189</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>189</td>
</tr>
<tr>
<td>Request Parameters</td>
<td>189</td>
</tr>
<tr>
<td>Response Syntax</td>
<td>189</td>
</tr>
<tr>
<td>Response Elements</td>
<td>190</td>
</tr>
<tr>
<td>Errors</td>
<td>189</td>
</tr>
<tr>
<td>See Also</td>
<td>190</td>
</tr>
<tr>
<td>SearchGameSessions</td>
<td>192</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>193</td>
</tr>
<tr>
<td>Request Parameters</td>
<td>193</td>
</tr>
<tr>
<td>Response Syntax</td>
<td>195</td>
</tr>
<tr>
<td>Response Elements</td>
<td>195</td>
</tr>
<tr>
<td>Errors</td>
<td>195</td>
</tr>
<tr>
<td>Examples</td>
<td>196</td>
</tr>
<tr>
<td>See Also</td>
<td>200</td>
</tr>
<tr>
<td>StartFleetActions</td>
<td>201</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>201</td>
</tr>
<tr>
<td>Request Parameters</td>
<td>201</td>
</tr>
<tr>
<td>Response Elements</td>
<td>202</td>
</tr>
<tr>
<td>Errors</td>
<td>202</td>
</tr>
<tr>
<td>Example</td>
<td>202</td>
</tr>
<tr>
<td>See Also</td>
<td>203</td>
</tr>
<tr>
<td>StartGameSessionPlacement</td>
<td>204</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>204</td>
</tr>
<tr>
<td>Request Parameters</td>
<td>205</td>
</tr>
<tr>
<td>Response Syntax</td>
<td>206</td>
</tr>
<tr>
<td>Response Elements</td>
<td>207</td>
</tr>
<tr>
<td>Errors</td>
<td>207</td>
</tr>
<tr>
<td>Example</td>
<td>208</td>
</tr>
<tr>
<td>See Also</td>
<td>209</td>
</tr>
<tr>
<td>StartMatchBackfill</td>
<td>210</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>210</td>
</tr>
<tr>
<td>Request Parameters</td>
<td>211</td>
</tr>
<tr>
<td>Response Syntax</td>
<td>212</td>
</tr>
<tr>
<td>Response Elements</td>
<td>212</td>
</tr>
<tr>
<td>Errors</td>
<td>213</td>
</tr>
<tr>
<td>See Also</td>
<td>213</td>
</tr>
<tr>
<td>StartMatchmaking</td>
<td>214</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>215</td>
</tr>
<tr>
<td>Request Parameters</td>
<td>215</td>
</tr>
<tr>
<td>Response Syntax</td>
<td>216</td>
</tr>
<tr>
<td>Response Elements</td>
<td>217</td>
</tr>
<tr>
<td>Errors</td>
<td>217</td>
</tr>
<tr>
<td>See Also</td>
<td>217</td>
</tr>
<tr>
<td>StopFleetActions</td>
<td>219</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>219</td>
</tr>
<tr>
<td>Request Parameters</td>
<td>219</td>
</tr>
<tr>
<td>Response Elements</td>
<td>219</td>
</tr>
<tr>
<td>Errors</td>
<td>219</td>
</tr>
<tr>
<td>Example</td>
<td>220</td>
</tr>
<tr>
<td>See Also</td>
<td>221</td>
</tr>
<tr>
<td>StopGameSessionPlacement</td>
<td>222</td>
</tr>
<tr>
<td>Request Syntax</td>
<td>222</td>
</tr>
<tr>
<td>Contents</td>
<td>287</td>
</tr>
<tr>
<td>-------------------------------------------------------------------------</td>
<td>-----</td>
</tr>
<tr>
<td>See Also</td>
<td>288</td>
</tr>
<tr>
<td>GameProperty</td>
<td>289</td>
</tr>
<tr>
<td>Contents</td>
<td>289</td>
</tr>
<tr>
<td>See Also</td>
<td>289</td>
</tr>
<tr>
<td>GameSession</td>
<td>290</td>
</tr>
<tr>
<td>Contents</td>
<td>290</td>
</tr>
<tr>
<td>See Also</td>
<td>293</td>
</tr>
<tr>
<td>GameSessionConnectionInfo</td>
<td>294</td>
</tr>
<tr>
<td>Contents</td>
<td>294</td>
</tr>
<tr>
<td>See Also</td>
<td>294</td>
</tr>
<tr>
<td>GameSessionDetail</td>
<td>296</td>
</tr>
<tr>
<td>Contents</td>
<td>296</td>
</tr>
<tr>
<td>See Also</td>
<td>296</td>
</tr>
<tr>
<td>GameSessionPlacement</td>
<td>297</td>
</tr>
<tr>
<td>Contents</td>
<td>297</td>
</tr>
<tr>
<td>See Also</td>
<td>300</td>
</tr>
<tr>
<td>GameSessionQueue</td>
<td>301</td>
</tr>
<tr>
<td>Contents</td>
<td>301</td>
</tr>
<tr>
<td>See Also</td>
<td>302</td>
</tr>
<tr>
<td>GameSessionQueueDestination</td>
<td>303</td>
</tr>
<tr>
<td>Contents</td>
<td>303</td>
</tr>
<tr>
<td>See Also</td>
<td>303</td>
</tr>
<tr>
<td>Instance</td>
<td>304</td>
</tr>
<tr>
<td>Contents</td>
<td>304</td>
</tr>
<tr>
<td>See Also</td>
<td>305</td>
</tr>
<tr>
<td>InstanceAccess</td>
<td>306</td>
</tr>
<tr>
<td>Contents</td>
<td>306</td>
</tr>
<tr>
<td>See Also</td>
<td>306</td>
</tr>
<tr>
<td>InstanceCredentials</td>
<td>308</td>
</tr>
<tr>
<td>Contents</td>
<td>308</td>
</tr>
<tr>
<td>See Also</td>
<td>308</td>
</tr>
<tr>
<td>IpPermission</td>
<td>309</td>
</tr>
<tr>
<td>Contents</td>
<td>309</td>
</tr>
<tr>
<td>See Also</td>
<td>309</td>
</tr>
<tr>
<td>MatchedPlayerSession</td>
<td>311</td>
</tr>
<tr>
<td>Contents</td>
<td>311</td>
</tr>
<tr>
<td>See Also</td>
<td>311</td>
</tr>
<tr>
<td>MatchmakingConfiguration</td>
<td>312</td>
</tr>
<tr>
<td>Contents</td>
<td>312</td>
</tr>
<tr>
<td>See Also</td>
<td>314</td>
</tr>
<tr>
<td>MatchmakingRuleSet</td>
<td>315</td>
</tr>
<tr>
<td>Contents</td>
<td>315</td>
</tr>
<tr>
<td>See Also</td>
<td>316</td>
</tr>
<tr>
<td>MatchmakingTicket</td>
<td>317</td>
</tr>
<tr>
<td>Contents</td>
<td>317</td>
</tr>
<tr>
<td>See Also</td>
<td>319</td>
</tr>
<tr>
<td>PlacedPlayerSession</td>
<td>320</td>
</tr>
<tr>
<td>Contents</td>
<td>320</td>
</tr>
<tr>
<td>See Also</td>
<td>320</td>
</tr>
<tr>
<td>Player</td>
<td>321</td>
</tr>
<tr>
<td>Contents</td>
<td>321</td>
</tr>
<tr>
<td>See Also</td>
<td>322</td>
</tr>
<tr>
<td>PlayerLatency</td>
<td>323</td>
</tr>
<tr>
<td>Contents</td>
<td>323</td>
</tr>
<tr>
<td>See Also</td>
<td>323</td>
</tr>
<tr>
<td>PlayerLatencyPolicy</td>
<td>324</td>
</tr>
</tbody>
</table>
Welcome

Amazon GameLift is a managed service for developers who need a scalable, dedicated server solution for their multiplayer games. Use Amazon GameLift for these tasks: (1) set up computing resources and deploy your game servers, (2) run game sessions and get players into games, (3) automatically scale your resources to meet player demand and manage costs, and (4) track in-depth metrics on game server performance and player usage.

The Amazon GameLift service API includes two important function sets:

- **Manage game sessions and player access** -- Retrieve information on available game sessions; create new game sessions; send player requests to join a game session.
- **Configure and manage game server resources** -- Manage builds, fleets, queues, and aliases; set auto-scaling policies; retrieve logs and metrics.

This reference guide describes the low-level service API for Amazon GameLift. You can use the API functionality with these tools:

- The Amazon Web Services software development kit (AWS SDK) is available in multiple languages including C++ and C#. Use the SDK to access the API programmatically from an application, such as a game client.
- The AWS command-line interface (CLI) tool is primarily useful for handling administrative actions, such as setting up and managing Amazon GameLift settings and resources. You can use the AWS CLI to manage all of your AWS services.
- The AWS Management Console for Amazon GameLift provides a web interface to manage your Amazon GameLift settings and resources. The console includes a dashboard for tracking key resources, including builds and fleets, and displays usage and performance metrics for your games as customizable graphs.
- Amazon GameLift Local is a tool for testing your game's integration with Amazon GameLift before deploying it on the service. This tools supports a subset of key API actions, which can be called from either the AWS CLI or programmatically. See **Testing an Integration**.

Learn more

- **Developer Guide** -- Read about Amazon GameLift features and how to use them.
- **Tutorials** -- Get started fast with walkthroughs and sample projects.
- **GameDev Blog** -- Stay up to date with new features and techniques.
- **GameDev Forums** -- Connect with the GameDev community.
- **Release notes** and **document history** -- Stay current with updates to the Amazon GameLift service, SDKs, and documentation.

**API SUMMARY**

This list offers a functional overview of the Amazon GameLift service API.

**Managing Games and Players**

Use these actions to start new game sessions, find existing game sessions, track game session status and other information, and enable player access to game sessions.

- **Discover existing game sessions**
  - **SearchGameSessions** (p. 192) -- Retrieve all available game sessions or search for game sessions that match a set of criteria.
• **Start new game sessions**
  - Start new games with Queues to find the best available hosting resources across multiple regions, minimize player latency, and balance game session activity for efficiency and cost effectiveness.
  - **StartGameSessionPlacement (p. 204)** -- Request a new game session placement and add one or more players to it.
  - **DescribeGameSessionPlacement (p. 117)** -- Get details on a placement request, including status.
  - **StopGameSessionPlacement (p. 222)** -- Cancel a placement request.
  - **CreateGameSession (p. 31)** -- Start a new game session on a specific fleet. *Available in Amazon GameLift Local.*

• **Match players to game sessions with FlexMatch matchmaking**
  - **StartMatchmaking (p. 214)** -- Request matchmaking for one players or a group who want to play together.
  - **StartMatchBackfill (p. 210)** -- Request additional player matches to fill empty slots in an existing game session.
  - **DescribeMatchmaking (p. 131)** -- Get details on a matchmaking request, including status.
  - **AcceptMatch (p. 7)** -- Register that a player accepts a proposed match, for matches that require player acceptance.
  - **StopMatchmaking (p. 225)** -- Cancel a matchmaking request.

• **Manage game session data**
  - **DescribeGameSessions (p. 123)** -- Retrieve metadata for one or more game sessions, including length of time active and current player count. *Available in Amazon GameLift Local.*
  - **DescribeGameSessionDetails (p. 113)** -- Retrieve metadata and the game session protection setting for one or more game sessions.
  - **UpdateGameSession (p. 246)** -- Change game session settings, such as maximum player count and join policy.
  - **GetGameSessionLogUrl (p. 157)** -- Get the location of saved logs for a game session.

• **Manage player sessions**
  - **CreatePlayerSession (p. 49)** -- Send a request for a player to join a game session. *Available in Amazon GameLift Local.*
  - **CreatePlayerSessions (p. 52)** -- Send a request for multiple players to join a game session. *Available in Amazon GameLift Local.*
  - **DescribePlayerSessions (p. 140)** -- Get details on player activity, including status, playing time, and player data. *Available in Amazon GameLift Local.*

**Setting Up and Managing Game Servers**

When setting up Amazon GameLift resources for your game, you first create a game build and upload it to Amazon GameLift. You can then use these actions to configure and manage a fleet of resources to run your game servers, scale capacity to meet player demand, access performance and utilization metrics, and more.

• **Manage game builds**
  - **CreateBuild (p. 13)** -- Create a new build using files stored in an Amazon S3 bucket. To create a build and upload files from a local path, use the AWS CLI command `upload-build`.
  - **ListBuilds (p. 168)** -- Get a list of all builds uploaded to a Amazon GameLift region.
  - **DescribeBuild (p. 87)** -- Retrieve information associated with a build.
  - **UpdateBuild (p. 230)** -- Change build metadata, including build name and version.
  - **DeleteBuild (p. 67)** -- Remove a build from Amazon GameLift.

• **Manage fleets**
  - **CreateFleet (p. 19)** -- Configure and activate a new fleet to run a build's game servers.

*API Version 2015-10-01*
- **ListFleets (p. 173)** -- Get a list of all fleet IDs in a Amazon GameLift region (all statuses).
- **DeleteFleet (p. 70)** -- Terminate a fleet that is no longer running game servers or hosting players.
- **View / update fleet configurations.**
  - **DescribeFleetAttributes (p. 93) / UpdateFleetAttributes (p. 234)** -- View or change a fleet's metadata and settings for game session protection and resource creation limits.
  - **DescribeFleetPortSettings (p. 107) / UpdateFleetPortSettings (p. 242)** -- View or change the inbound permissions (IP address and port setting ranges) allowed for a fleet.
  - **DescribeRuntimeConfiguration (p. 144) / UpdateRuntimeConfiguration (p. 258)** -- View or change what server processes (and how many) to run on each instance in a fleet.
- **Control fleet capacity**
  - **DescribeEC2InstanceLimits (p. 90)** -- Retrieve maximum number of instances allowed for the current AWS account and the current usage level.
  - **DescribeFleetCapacity (p. 97) / UpdateFleetCapacity (p. 238)** -- Retrieve the capacity settings and the current number of instances in a fleet; adjust fleet capacity settings to scale up or down.
  - **Autoscale** -- Manage auto-scaling rules and apply them to a fleet.
    - **PutScalingPolicy (p. 178)** -- Create a new auto-scaling policy, or update an existing one.
    - **DescribeScalingPolicies (p. 147)** -- Retrieve an existing auto-scaling policy.
    - **DeleteScalingPolicy (p. 77)** -- Delete an auto-scaling policy and stop it from affecting a fleet's capacity.
  - **StartFleetActions (p. 201)** -- Restart a fleet's auto-scaling policies.
  - **StopFleetActions (p. 219)** -- Suspend a fleet's auto-scaling policies.
- **Manage game session queues**
  - **CreateGameSessionQueue (p. 37)** -- Create a queue for processing requests for new game sessions.
  - **DescribeGameSessionQueues (p. 120)** -- Retrieve game session queues defined in a Amazon GameLift region.
  - **UpdateGameSessionQueue (p. 250)** -- Change the configuration of a game session queue.
  - **DeleteGameSessionQueue (p. 73)** -- Remove a game session queue from the region.
- **Manage FlexMatch resources**
  - **CreateMatchmakingConfiguration (p. 41)** -- Create a matchmaking configuration with instructions for building a player group and placing in a new game session.
  - **DescribeMatchmakingConfigurations (p. 134)** -- Retrieve matchmaking configurations defined a Amazon GameLift region.
  - **UpdateMatchmakingConfiguration (p. 253)** -- Change settings for matchmaking configuration.
  - **DeleteMatchmakingConfiguration (p. 75)** -- Remove a matchmaking configuration from the region.
  - **CreateMatchmakingRuleSet (p. 46)** -- Create a set of rules to use when searching for player matches.
  - **DescribeMatchmakingRuleSets (p. 137)** -- Retrieve matchmaking rule sets defined in a Amazon GameLift region.
  - **ValidateMatchmakingRuleSet (p. 262)** -- Verify syntax for a set of matchmaking rules.
- **Access fleet activity statistics**
  - **DescribeFleetUtilization (p. 110)** -- Get current data on the number of server processes, game sessions, and players currently active on a fleet.
  - **DescribeFleetEvents (p. 101)** -- Get a fleet's logged events for a specified time span.
  - **DescribeGameSessions (p. 123)** -- Retrieve metadata associated with one or more game sessions, including length of time active and current player count.
- **Remotely access an instance**
• **DescribeInstances** (p. 127) -- Get information on each instance in a fleet, including instance ID, IP address, and status.
• **GetInstanceAccess** (p. 160) -- Request access credentials needed to remotely connect to a specified instance in a fleet.

• **Manage fleet aliases**
  • **CreateAlias** (p. 10) -- Define a new alias and optionally assign it to a fleet.
  • **ListAliases** (p. 165) -- Get all fleet aliases defined in an Amazon GameLift region.
  • **DescribeAlias** (p. 84) -- Retrieve information on an existing alias.
  • **UpdateAlias** (p. 227) -- Change settings for a alias, such as redirecting it from one fleet to another.
  • **DeleteAlias** (p. 65) -- Remove an alias from the region.
  • **ResolveAlias** (p. 189) -- Get the fleet ID that a specified alias points to.

• **Manage VPC peering connections for fleets**
  • **CreateVpcPeeringAuthorization** (p. 56) -- Authorize a peering connection to one of your VPCs.
  • **DescribeVpcPeeringAuthorizations** (p. 152) -- Retrieve valid peering connection authorizations.
  • **DeleteVpcPeeringAuthorization** (p. 80) -- Delete a peering connection authorization.
  • **CreateVpcPeeringConnection** (p. 61) -- Establish a peering connection between the VPC for an Amazon GameLift fleet and one of your VPCs.
  • **DescribeVpcPeeringConnections** (p. 154) -- Retrieve information on active or pending VPC peering connections with a Amazon GameLift fleet.
  • **DeleteVpcPeeringConnection** (p. 82) -- Delete a VPC peering connection with a Amazon GameLift fleet.

This document was last published on June 21, 2018.
Actions

The following actions are supported:

- AcceptMatch (p. 7)
- CreateAlias (p. 10)
- CreateBuild (p. 13)
- CreateFleet (p. 19)
- CreateGameSession (p. 31)
- CreateGameSessionQueue (p. 37)
- CreateMatchmakingConfiguration (p. 41)
- CreateMatchmakingRuleSet (p. 46)
- CreatePlayerSession (p. 49)
- CreatePlayerSessions (p. 52)
- CreateVpcPeeringAuthorization (p. 56)
- CreateVpcPeeringConnection (p. 61)
- DeleteAlias (p. 65)
- DeleteBuild (p. 67)
- DeleteFleet (p. 70)
- DeleteGameSessionQueue (p. 73)
- DeleteMatchmakingConfiguration (p. 75)
- DeleteScalingPolicy (p. 77)
- DeleteVpcPeeringAuthorization (p. 80)
- DeleteVpcPeeringConnection (p. 82)
- DescribeAlias (p. 84)
- DescribeBuild (p. 87)
- DescribeEC2InstanceLimits (p. 90)
- DescribeFleetAttributes (p. 93)
- DescribeFleetCapacity (p. 97)
- DescribeFleetEvents (p. 101)
- DescribeFleetPortSettings (p. 107)
- DescribeFleetUtilization (p. 110)
- DescribeGameSessionDetails (p. 113)
- DescribeGameSessionPlacement (p. 117)
- DescribeGameSessionQueues (p. 120)
- DescribeGameSessions (p. 123)
- DescribeInstances (p. 127)
- DescribeMatchmaking (p. 131)
- DescribeMatchmakingConfigurations (p. 134)
- DescribeMatchmakingRuleSets (p. 137)
- DescribePlayerSessions (p. 140)
- DescribeRuntimeConfiguration (p. 144)
- DescribeScalingPolicies (p. 147)
- DescribeVpcPeeringAuthorizations (p. 152)
• DescribeVpcPeeringConnections (p. 154)
• GetGameSessionLogUrl (p. 157)
• GetInstanceAccess (p. 160)
• ListAliases (p. 165)
• ListBuilds (p. 168)
• ListFleets (p. 173)
• PutScalingPolicy (p. 178)
• RequestUploadCredentials (p. 186)
• ResolveAlias (p. 189)
• SearchGameSessions (p. 192)
• StartFleetActions (p. 201)
• StartGameSessionPlacement (p. 204)
• StartMatchBackfill (p. 210)
• StartMatchmaking (p. 214)
• StopFleetActions (p. 219)
• StopGameSessionPlacement (p. 222)
• StopMatchmaking (p. 225)
• UpdateAlias (p. 227)
• UpdateBuild (p. 230)
• UpdateFleetAttributes (p. 234)
• UpdateFleetCapacity (p. 238)
• UpdateFleetPortSettings (p. 242)
• UpdateGameSession (p. 246)
• UpdateGameSessionQueue (p. 250)
• UpdateMatchmakingConfiguration (p. 253)
• UpdateRuntimeConfiguration (p. 258)
• ValidateMatchmakingRuleSet (p. 262)
AcceptMatch

AcceptMatch registers a player’s acceptance or rejection of a proposed FlexMatch match. A matchmaking configuration may require player acceptance; if so, then matches built with that configuration cannot be completed unless all players accept the proposed match within a specified time limit.

When FlexMatch builds a match, all the matchmaking tickets involved in the proposed match are placed into status `REQUIRES_ACCEPTANCE`. This is a trigger for your game to get acceptance from all players in the ticket. Acceptances are only valid for tickets when they are in this status; all other acceptances result in an error.

To register acceptance, specify the ticket ID, a response, and one or more players. Once all players have registered acceptance, the matchmaking tickets advance to status `PLACING`, where a new game session is created for the match.

If any player rejects the match, or if acceptances are not received before a specified timeout, the proposed match is dropped. The matchmaking tickets are then handled in one of two ways: For tickets where all players accepted the match, the ticket status is returned to `SEARCHING` to find a new match. For tickets where one or more players failed to accept the match, the ticket status is set to `FAILED`, and processing is terminated. A new matchmaking request for these players can be submitted as needed.

Matchmaking-related operations include:

- StartMatchmaking (p. 214)
- DescribeMatchmaking (p. 131)
- StopMatchmaking (p. 225)
- AcceptMatch (p. 7)
- StartMatchBackfill (p. 210)

Request Syntax

```
{
    "AcceptanceType": "string",
    "PlayerIds": [ "string" ],
    "TicketId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.

**AcceptanceType (p. 7)**

Player response to the proposed match.

Type: String

Valid Values: ACCEPT | REJECT
Required: Yes

**PlayerIds (p. 7)**

Unique identifier for a player delivering the response. This parameter can include one or multiple player IDs.

Type: Array of strings


Required: Yes

**TicketId (p. 7)**

Unique identifier for a matchmaking ticket. The ticket must be in status REQUIRES_ACCEPTANCE; otherwise this request will fail.

Type: String


Pattern: [a-zA-Z0-9-.\]+

Required: Yes

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

**Errors**

For information about the errors that are common to all actions, see Common Errors (p. 347).

**InternalServiceException**

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

**InvalidRequestException**

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

**NotFoundException**

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

**UnsupportedRegionException**

The requested operation is not supported in the region specified.

HTTP Status Code: 400
See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
CreateAlias

Creates an alias for a fleet. In most situations, you can use an alias ID in place of a fleet ID. By using a fleet alias instead of a specific fleet ID, you can switch gameplay and players to a new fleet without changing your game client or other game components. For example, for games in production, using an alias allows you to seamlessly redirect your player base to a new game server update.

Amazon GameLift supports two types of routing strategies for aliases: simple and terminal. A simple alias points to an active fleet. A terminal alias is used to display messaging or link to a URL instead of routing players to an active fleet. For example, you might use a terminal alias when a game version is no longer supported and you want to direct players to an upgrade site.

To create a fleet alias, specify an alias name, routing strategy, and optional description. Each simple alias can point to only one fleet, but a fleet can have multiple aliases. If successful, a new alias record is returned, including an alias ID, which you can reference when creating a game session. You can reassign an alias to another fleet by calling UpdateAlias.

Alias-related operations include:

- CreateAlias (p. 10)
- ListAliases (p. 165)
- DescribeAlias (p. 84)
- UpdateAlias (p. 227)
- DeleteAlias (p. 65)
- ResolveAlias (p. 189)

Request Syntax

```
{
   "Description": "string",
   "Name": "string",
   "RoutingStrategy": {
      "FleetId": "string",
      "Message": "string",
      "Type": "string"
   }
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.

**Name (p. 10)**

Descriptive label that is associated with an alias. Alias names do not need to be unique.

Type: String

Pattern: .*\S.*
Required: Yes
**RoutingStrategy (p. 10)**
Object that specifies the fleet and routing type to use for the alias.
Type: RoutingStrategy (p. 329) object
Required: Yes
**Description (p. 10)**
Human-readable description of an alias.
Type: String
Required: No

### Response Syntax

```
{
    "Alias": {
        "AliasArn": "string",
        "AliasId": "string",
        "CreationTime": number,
        "Description": "string",
        "LastUpdatedTime": number,
        "Name": "string",
        "RoutingStrategy": {
            "FleetId": "string",
            "Message": "string",
            "Type": "string"
        }
    }
}
```

### Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**Alias (p. 11)**
Object that describes the newly created alias record.
Type: Alias (p. 266) object

### Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

**ConflictException**

The requested operation would cause a conflict with the current state of a service resource associated with the request. Resolve the conflict before retrying this request.
HTTP Status Code: 400

**InternalServiceException**

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

**InvalidRequestException**

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

**LimitExceededException**

The requested operation would cause the resource to exceed the allowed service limit. Resolve the issue before retrying.

HTTP Status Code: 400

**UnauthorizedException**

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
CreateBuild

Creates a new Amazon GameLift build record for your game server binary files and points to the location of your game server build files in an Amazon Simple Storage Service (Amazon S3) location.

Game server binaries must be combined into a .zip file for use with Amazon GameLift. See Uploading Your Game for more information.

**Important**
To create new builds quickly and easily, use the AWS CLI command `upload-build`. This helper command uploads your build and creates a new build record in one step, and automatically handles the necessary permissions. See Upload Build Files to Amazon GameLift for more help.

The CreateBuild operation should be used only when you need to manually upload your build files, as in the following scenarios:

- Store a build file in an Amazon S3 bucket under your own AWS account. To use this option, you must first give Amazon GameLift access to that Amazon S3 bucket. See Create a Build with Files in Amazon S3 for detailed help. To create a new build record using files in your Amazon S3 bucket, call CreateBuild and specify a build name, operating system, and the storage location of your game build.

- Upload a build file directly to Amazon GameLift's Amazon S3 account. To use this option, you first call CreateBuild with a build name and operating system. This action creates a new build record and returns an Amazon S3 storage location (bucket and key only) and temporary access credentials. Use the credentials to manually upload your build file to the storage location (see the Amazon S3 topic Uploading Objects). You can upload files to a location only once.

If successful, this operation creates a new build record with a unique build ID and places it in INITIALIZED status. You can use DescribeBuild (p. 87) to check the status of your build. A build must be in READY status before it can be used to create fleets.

Build-related operations include:

- CreateBuild (p. 13)
- ListBuilds (p. 168)
- DescribeBuild (p. 87)
- UpdateBuild (p. 230)
- DeleteBuild (p. 67)

### Request Syntax

```json
{
   "Name": "string",
   "OperatingSystem": "string",
   "StorageLocation": {
      "Bucket": "string",
      "Key": "string",
      "RoleArn": "string"
   },
   "Version": "string"
}
```

API Version 2015-10-01
Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

Note
In the following list, the required parameters are described first.

Name (p. 13)

Descriptive label that is associated with a build. Build names do not need to be unique. You can use UpdateBuild (p. 230) to change this value later.

Type: String


Required: No

OperatingSystem (p. 13)

Operating system that the game server binaries are built to run on. This value determines the type of fleet resources that you can use for this build. If your game build contains multiple executables, they all must run on the same operating system. If an operating system is not specified when creating a build, Amazon GameLift uses the default value (WINDOWS_2012). This value cannot be changed later.

Type: String

Valid Values: WINDOWS_2012 | AMAZON_LINUX

Required: No

StorageLocation (p. 13)

Information indicating where your game build files are stored. Use this parameter only when creating a build with files stored in an Amazon S3 bucket that you own. The storage location must specify an Amazon S3 bucket name and key, as well as a role ARN that you set up to allow Amazon GameLift to access your Amazon S3 bucket. The S3 bucket must be in the same region that you want to create a new build in.

Type: S3Location (p. 333) object

Required: No

Version (p. 13)

Version that is associated with this build. Version strings do not need to be unique. You can use UpdateBuild (p. 230) to change this value later.

Type: String


Required: No

Response Syntax

```
{
    "Build": {
```
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**Build (p. 14)**

The newly created build record, including a unique build ID and status.

Type: Build (p. 271) object

**StorageLocation (p. 14)**

Amazon S3 location for your game build file, including bucket name and key.

Type: S3Location (p. 333) object

**UploadCredentials (p. 14)**

This element is returned only when the operation is called without a storage location. It contains credentials to use when you are uploading a build file to an Amazon S3 bucket that is owned by Amazon GameLift. Credentials have a limited life span. To refresh these credentials, call RequestUploadCredentials (p. 186).

Type: AwsCredentials (p. 270) object

**Errors**

For information about the errors that are common to all actions, see Common Errors (p. 347).

**ConflictException**

The requested operation would cause a conflict with the current state of a service resource associated with the request. Resolve the conflict before retrying this request.

HTTP Status Code: 400

**InternalServiceException**

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.
HTTP Status Code: 500
InvalidRequestException

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400
UnauthorizedException

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

Examples

Create a build with files in your own S3 bucket

This example creates a new build record using a build file that is stored in your own Amazon S3 bucket. The example assumes that you've already created a role that gives Amazon GameLift permission to access your bucket. Once the build file is successfully validated, the new build's status changes to READY. Notice that the example request does not specify an operating system; in this scenario, the new build defaults to WINDOWS_2012.

HTTP requests are authenticated using an AWS Signature Version 4 signature in the Authorization header field.

Sample Request

```
POST / HTTP/1.1
Host: gamelift.us-west-2.amazonaws.com;
Accept-Encoding: identity
Content-Length: 259
User-Agent: aws-cli/1.11.36 Python/2.7.9 Windows/7 botocore/1.4.93
Content-Type: application/x-amz-json-1.0
Authorization: AWS4-HMAC-SHA256  Credential=AKIAIOSFODNN7EXAMPLE/20170406/us-west-2/ gamelift/aws4_request, SignedHeaders=content-type;host;x-amz-date;x-amz-target,
    Signature=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
X-Amz-Date: 20170406T004805Z
X-Amz-Target: GameLift.CreateBuild
{
    "Name": "My_Game_Server_Build_One",
    "Version": "12345.678",
    "StorageLocation": {
        "Bucket": "my_game_build_files",
        "Key": "mygame_build_7.0.1, 7.0.2",
        "RoleArn": "arn:aws:iam::111122223333:role/GameLiftAccess"
    }
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: b34f8665-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 427
Date: Thu, 06 Apr 2017 00:48:07 GMT
```
Create a build and upload your game server files to Amazon GameLift's S3 bucket

This example creates a new build record AND gets a storage location and temporary credentials that allow you to manually upload a game build to Amazon GameLift's Amazon S3 account. Once you've successfully uploaded your build, the Amazon GameLift service is notified and the new build's status changes to READY.

HTTP requests are authenticated using an AWS Signature Version 4 signature in the Authorization header field.

Sample Request

POST / HTTP/1.1
Host: gamelift.us-west-2.amazonaws.com;
Accept-Encoding: identity
Content-Length: 109
User-Agent: aws-cli/1.11.36 Python/2.7.9 Windows/7 botocore/1.4.93
Content-Type: application/x-amz-json-1.0
Authorization: AWS4-HMAC-SHA256  Credential=AKIAIOSFODNN7EXAMPLE/20170406/us-west-2/
gamelift/aws4_request, SignedHeaders=content-type;host;x-amz-date;x-amz-target,
 Signature=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
X-Amz-Date: 20170406T004805Z
X-Amz-Target: GameLift.CreateBuild

{
    "Name": "My_Game_Server_Build_One",
    "Version": "12345.678",
    "OperatingSystem": "AMAZON_LINUX"
}

Sample Response

HTTP/1.1 200 OK
x-amzn-RequestId: b34f8665-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 677
Date: Thu, 06 Apr 2017 00:48:07 GMT

{
    "Build": {
    "BuildId": "build-1111aaaa-22bb-33cc-44dd-5555eeeee66ff",
    "CreationTime": 1496708916.18,
    "Name": "My_Game_Server_Build_One",
    "OperatingSystem": "WINDOWS_2012",
    "SizeOnDisk": 0,
    "Status": "INITIALIZED",
    "Version": "12345.678"
    },
    "StorageLocation": {
    "Bucket": "my_game_build_files",
    "Key": "mygame_build_7.0.1, 7.0.2"
    }
}
"OperatingSystem": "AMAZON_LINUX",
"SizeOnDisk": 0,
"Status": "INITIALIZED",
"Version": "12345.678"
},
"StorageLocation": {
"Bucket": "gamelift-builds-us-west-2",
"Key": "123456789012/build-1111aaaa-22bb-33cc-44dd-5555eeee66ff"
},
"UploadCredentials": {
"AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
"SecretAccessKey": "wJalrXUtnnFEMI/K7MDENG/bPxRfiICX=EXAMPLEKEY",
"SessionToken": "AgoGb3JpZ2luENZ...EXAMPLETOKEN=="
}
}

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
CreateFleet

Creates a new fleet to run your game servers. A fleet is a set of Amazon Elastic Compute Cloud (Amazon EC2) instances, each of which can run multiple server processes to host game sessions. You set up a fleet to use instances with certain hardware specifications (see Amazon EC2 Instance Types for more information), and deploy your game build to run on each instance.

To create a new fleet, you must specify the following: (1) a fleet name, (2) the build ID of a successfully uploaded game build, (3) an EC2 instance type, and (4) a run-time configuration, which describes the server processes to run on each instance in the fleet. If you don’t specify a fleet type (on-demand or spot), the new fleet uses on-demand instances by default.

You can also configure the new fleet with the following settings:

- Fleet description
- Access permissions for inbound traffic
- Fleet-wide game session protection
- Resource usage limits
- VPC peering connection (see VPC Peering with Amazon GameLift Fleets)

If you use Amazon CloudWatch for metrics, you can add the new fleet to a metric group. By adding multiple fleets to a metric group, you can view aggregated metrics for all the fleets in the group.

If the CreateFleet call is successful, Amazon GameLift performs the following tasks. You can track the process of a fleet by checking the fleet status or by monitoring fleet creation events:

- Creates a fleet record. Status: NEW.
- Begins writing events to the fleet event log, which can be accessed in the Amazon GameLift console.
  - Sets the fleet's target capacity to 1 (desired instances), which triggers Amazon GameLift to start one new EC2 instance.
  - Downloads the game build to the new instance and installs it. Statuses: DOWNLOADING, VALIDATING, BUILDING.
  - Starts launching server processes on the instance. If the fleet is configured to run multiple server processes per instance, Amazon GameLift staggers each launch by a few seconds. Status: ACTIVATING.
  - Sets the fleet's status to ACTIVE as soon as one server process is ready to host a game session.

Fleet-related operations include:

- CreateFleet (p. 19)
- ListFleets (p. 173)
- DeleteFleet (p. 70)
- Describe fleets:
  - DescribeFleetAttributes (p. 93)
  - DescribeFleetCapacity (p. 97)
  - DescribeFleetPortSettings (p. 107)
  - DescribeFleetUtilization (p. 110)
  - DescribeRuntimeConfiguration (p. 144)
  - DescribeEC2InstanceLimits (p. 90)
  - DescribeFleetEvents (p. 101)
• Update fleets:
  • UpdateFleetAttributes (p. 234)
  • UpdateFleetCapacity (p. 238)
  • UpdateFleetPortSettings (p. 242)
  • UpdateRuntimeConfiguration (p. 258)
• Manage fleet actions:
  • StartFleetActions (p. 201)
  • StopFleetActions (p. 219)

Request Syntax

```json
{
  "BuildId": "string",
  "Description": "string",
  "EC2InboundPermissions": [
    {
      "FromPort": number,
      "IpRange": "string",
      "Protocol": "string",
      "ToPort": number
    }
  ],
  "EC2InstanceType": "string",
  "FleetType": "string",
  "LogPaths": [ "string" ],
  "MetricGroups": [ "string" ],
  "Name": "string",
  "NewGameSessionProtectionPolicy": "string",
  "PeerVpcAwsAccountId": "string",
  "PeerVpcId": "string",
  "ResourceCreationLimitPolicy": {
    "NewGameSessionsPerCreator": number,
    "PolicyPeriodInMinutes": number
  },
  "RuntimeConfiguration": {
    "GameSessionActivationTimeoutSeconds": number,
    "MaxConcurrentGameSessionActivation": number,
    "ServerLaunchParameters": [ {
      "ConcurrentExecutions": number,
      "LaunchPath": "string",
      "Parameters": "string"
    } ],
    "ServerLaunchPath": "string"
  }
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**
In the following list, the required parameters are described first.
**BuildId (p. 20)**

Unique identifier for a build to be deployed on the new fleet. The build must have been successfully uploaded to Amazon GameLift and be in a READY status. This fleet setting cannot be changed once the fleet is created.

Type: String

Pattern: `^build-\S+$`

Required: Yes

**EC2InstanceType (p. 20)**

Name of an EC2 instance type that is supported in Amazon GameLift. A fleet instance type determines the computing resources of each instance in the fleet, including CPU, memory, storage, and networking capacity. Amazon GameLift supports the following EC2 instance types. See Amazon EC2 Instance Types for detailed descriptions.

Type: String

Valid Values: t2.micro | t2.small | t2.medium | t2.large | c3.large | c3.xlarge | c3.2xlarge | c3.4xlarge | c3.8xlarge | c4.large | c4.xlarge | c4.2xlarge | c4.4xlarge | c4.8xlarge | r3.large | r3.xlarge | r3.2xlarge | r3.4xlarge | r3.8xlarge | r4.large | r4.xlarge | r4.2xlarge | r4.4xlarge | r4.8xlarge | r4.16xlarge | m3.medium | m3.large | m3.xlarge | m3.2xlarge | m4.large | m4.xlarge | m4.2xlarge | m4.4xlarge | m4.10xlarge

Required: Yes

**Name (p. 20)**

Descriptive label that is associated with a fleet. Fleet names do not need to be unique.

Type: String


Required: Yes

**Description (p. 20)**

Human-readable description of a fleet.

Type: String


Required: No

**EC2InboundPermissions (p. 20)**

Range of IP addresses and port settings that permit inbound traffic to access server processes running on the fleet. If no inbound permissions are set, including both IP address range and port range, the server processes in the fleet cannot accept connections. You can specify one or more sets of permissions for a fleet.

Type: Array of IpPermission (p. 309) objects

Array Members: Maximum number of 50 items.

Required: No
FleetType (p. 20)

Indicates whether to use on-demand instances or spot instances for this fleet. If empty, the default is ON_DEMAND. Both categories of instances use identical hardware and configurations, based on the instance type selected for this fleet. You can acquire on-demand instances at any time for a fixed price and keep them as long as you need them. Spot instances have lower prices, but spot pricing is variable, and while in use they can be interrupted (with a two-minute notification). Learn more about Amazon GameLift spot instances with at Choose Computing Resources.

Type: String

Valid Values: ON_DEMAND | SPOT

Required: No

LogPaths (p. 20)

This parameter is no longer used. Instead, to specify where Amazon GameLift should store log files once a server process shuts down, use the Amazon GameLift server API ProcessReady() and specify one or more directory paths in logParameters. See more information in the Server API Reference.

Type: Array of strings


Required: No

MetricGroups (p. 20)

Name of a metric group to add this fleet to. A metric group tracks metrics across all fleets in the group. Use an existing metric group name to add this fleet to the group, or use a new name to create a new metric group. A fleet can only be included in one metric group at a time.

Type: Array of strings

Array Members: Maximum number of 1 item.

Length Constraints: Minimum length of 1. Maximum length of 255.

Required: No

NewGameSessionProtectionPolicy (p. 20)

Game session protection policy to apply to all instances in this fleet. If this parameter is not set, instances in this fleet default to no protection. You can change a fleet's protection policy using UpdateFleetAttributes (p. 234), but this change will only affect sessions created after the policy change. You can also set protection for individual instances using UpdateGameSession (p. 246).

- NoProtection -- The game session can be terminated during a scale-down event.
- FullProtection -- If the game session is in an ACTIVE status, it cannot be terminated during a scale-down event.

Type: String

Valid Values: NoProtection | FullProtection

Required: No

PeerVpcAwsAccountId (p. 20)

Unique identifier for the AWS account with the VPC that you want to peer your Amazon GameLift fleet with. You can find your Account ID in the AWS Management Console under account settings.
Request Parameters

Type: String
Required: No

PeerVpcId (p. 20)

Unique identifier for a VPC with resources to be accessed by your Amazon GameLift fleet. The VPC must be in the same region where your fleet is deployed. To get VPC information, including IDs, use the Virtual Private Cloud service tools, including the VPC Dashboard in the AWS Management Console.

Type: String
Required: No

ResourceCreationLimitPolicy (p. 20)

Policy that limits the number of game sessions an individual player can create over a span of time for this fleet.

Type: ResourceCreationLimitPolicy (p. 328) object
Required: No

RuntimeConfiguration (p. 20)

Instructions for launching server processes on each instance in the fleet. The run-time configuration for a fleet has a collection of server process configurations, one for each type of server process to run on an instance. A server process configuration specifies the location of the server executable, launch parameters, and the number of concurrent processes with that configuration to maintain on each instance. A CreateFleet request must include a run-time configuration with at least one server process configuration; otherwise the request fails with an invalid request exception. (This parameter replaces the parameters ServerLaunchPath and ServerLaunchParameters; requests that contain values for these parameters instead of a run-time configuration will continue to work.)

Type: RuntimeConfiguration (p. 331) object
Required: No

ServerLaunchParameters (p. 20)

This parameter is no longer used. Instead, specify server launch parameters in the RuntimeConfiguration parameter. (Requests that specify a server launch path and launch parameters instead of a run-time configuration will continue to work.)

Type: String
Required: No

ServerLaunchPath (p. 20)

This parameter is no longer used. Instead, specify a server launch path using the RuntimeConfiguration parameter. (Requests that specify a server launch path and launch parameters instead of a run-time configuration will continue to work.)

Type: String
Response Syntax

```json
{
   "FleetAttributes": {
      "BuildId": "string",
      "CreationTime": number,
      "Description": "string",
      "FleetArn": "string",
      "FleetId": "string",
      "FleetType": "string",
      "InstanceType": "string",
      "LogPaths": [ "string" ],
      "MetricGroups": [ "string" ],
      "Name": "string",
      "NewGameSessionProtectionPolicy": "string",
      "OperatingSystem": "string",
      "ResourceCreationLimitPolicy": {
         "NewGameSessionsPerCreator": number,
         "PolicyPeriodInMinutes": number
      },
      "ServerLaunchParameters": "string",
      "ServerLaunchPath": "string",
      "Status": "string",
      "StoppedActions": [ "string" ],
      "TerminationTime": number
   }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**FleetAttributes (p. 24)**

Properties for the newly created fleet.

Type: FleetAttributes (p. 280) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

**ConflictException**

The requested operation would cause a conflict with the current state of a service resource associated with the request. Resolve the conflict before retrying this request.

HTTP Status Code: 400

**InternalServiceException**

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.
HTTP Status Code: 500
InvalidRequestException
One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400
LimitExceededException
The requested operation would cause the resource to exceed the allowed service limit. Resolve the issue before retrying.

HTTP Status Code: 400
NotFoundException
A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400
UnauthorizedException
The client failed authentication. Clients should not retry such requests.

Examples
Create fleet with minimal configuration (Linux)

This example creates a fleet for a game server build with a minimal configuration and a placeholder launch path. You can use this fleet to create queues and matchmakers, test Amazon GameLift Server API calls, etc. Once you're ready to start hosting game sessions, complete the configuration settings with the UpdateFleet operations.

HTTP requests are authenticated using an AWS Signature Version 4 signature in the Authorization header field.

Sample Request

```plaintext
POST / HTTP/1.1
Host: gamelift.us-west-2.amazonaws.com;
Accept-Encoding: identity
Content-Length: 338
User-Agent: aws-cli/1.11.36 Python/2.7.9 Windows/7 botocore/1.4.93
Content-Type: application/x-amz-json-1.0
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20170406/us-west-2/
gamelift/aws4_request, SignedHeaders=content-type;host;x-amz-date;x-amz-target,
Signature=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
X-Amz-Date: 20170406T004805Z
X-Amz-Target: GameLift.CreateFleet

{
    "Name": "My_Fleet_1",
    "Description": "A minimal sample fleet",
    "BuildId": "build-1111aaaa-22bb-33cc-44dd-5555eeee66ff",
    "EC2InstanceType": "c4.large",
    "FleetType": "ON_DEMAND",
```
"RuntimeConfiguration": {
    "ServerProcesses": [
        {
            "LaunchPath": "C:/local/game/mygame.exe",
            "ConcurrentExecutions": 1
        }
    ]
}

CLI syntax:

```
$aws gamelift create-fleet --name My_Fleet_2
    --description "Another minimal sample fleet"
    --build-id build-1111aaaaa-22bb-33cc-44dd-5555eee66ff
    --ec2-instance-type "c4.large"
    --fleet-type "ON-DEMAND"
    --runtime-configuration "ServerProcesses=[{LaunchPath=/local/game/mygame.exe,ConcurrentExecutions=1}]"
```

**Sample Response**

HTTP/1.1 200 OK
x-amzn-RequestId: b34f8665-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 607
Date: Thu, 06 Apr 2017 00:48:07 GMT

```
{
    "FleetAttributes": {
        "BuildId": "build-1111aaaaa-22bb-33cc-44dd-5555eee66ff",
        "CreationTime": 1496365885.44,
        "Description": "A minimal sample fleet",
        "FleetArn": "arn:aws:gamelift:us-west-2::fleet/fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
        "FleetId": "fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
        "FleetType": "ON-DEMAND",
        "MetricGroups": ["default"],
        "Name": "My_Fleet_1",
        "NewGameSessionProtectionPolicy": "NoProtection",
        "OperatingSystem": "AMAZON_LINUX",
        "ServerLaunchPath": "C:/local/game/mygame.exe",
        "Status": "NEW"
    }
}
```

**Create fleet with minimal configuration (Windows)**

This example creates a fleet for a game server build with a minimal configuration and a placeholder launch path. You can use this fleet to create queues and matchmakers, test Amazon GameLift Server API calls, etc. Once you're ready to start hosting game sessions, complete the configuration settings with the UpdateFleet operations.

HTTP requests are authenticated using an AWS Signature Version 4 signature in the Authorization header field.

**Sample Request**

```
POST / HTTP/1.1
Host: gamelift.us-west-2.amazonaws.com;
Accept-Encoding: identity
Content-Length: 336
User-Agent: aws-cli/1.11.36 Python/2.7.9 Windows/7 botocore/1.4.93
```
Amazon GameLift Service API Reference
Examples

Content-Type: application/x-amz-json-1.0
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20170406/us-west-2/gamelift/aws4_request, SignedHeaders=content-type;host;x-amz-date;x-amz-target, Signature=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
X-Amz-Date: 20170406T004805Z
X-Amz-Target: GameLift.CreateFleet

{
   "Name": "My_Fleet_1",
   "Description": "A minimal sample fleet",
   "BuildId": "build-1111aaaa-22bb-33cc-44dd-5555eeee66ff",
   "EC2InstanceType": "c4.large",
   "FleetType": "ON_DEMAND",
   "RuntimeConfiguration": {
      "ServerProcesses": [
         {"LaunchPath": "c:\game\mygame.exe",
          "ConcurrentExecutions": 1}
      ]
   }
}

CLI syntax:
$aws gamelift create-fleet --name My_Fleet_2
--description "Another minimal sample fleet"
--build-id build-1111aaaa-22bb-33cc-44dd-5555eeee66ff
--ec2-instance-type "c4.large"
--fleet-type "ON-DEMAND"
--runtime-configuration "ServerProcesses=[{LaunchPath=c:\game
\mygame.exe,ConcurrentExecutions=1}]"

Sample Response

HTTP/1.1 200 OK
X-amzn-RequestId: b34f8665-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 600
Date: Thu, 06 Apr 2017 00:48:07 GMT

{
   "FleetAttributes": {
      "BuildId": "build-1111aaaa-22bb-33cc-44dd-5555eeee66ff",
      "CreationTime": 1496365885.44,
      "Description": "A minimal sample fleet",
      "FleetArn": "arn:aws:gamelift:us-west-2::fleet/fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
      "FleetId": "fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
      "FleetType": "ON_DEMAND",
      "MetricGroups": ["default"],
      "Name": "My_Fleet_1",
      "NewGameSessionProtectionPolicy": "NoProtection",
      "OperatingSystem": "WINDOWS_2012",
      "ServerLaunchPath": "c:\game\mygame.exe",
      "Status": "NEW"
   }
}

Create fleet with full configuration

This example creates a new fleet with complete configuration details. In this example, the runtime configuration sets up two different server process configurations. The first configuration calls for three concurrent processes of the game server to run in standard mode. The second configuration calls for one process of the game server to run in a test mode. As a result, each instance in this fleet is configured
to run four processes of the game server concurrently. This example also requests that a VPC peering connection be established between the VPC for this new fleet and another VPC that has been pre-authorized. The game server build in this example is Windows.

HTTP requests are authenticated using an AWS Signature Version 4 signature in the Authorization header field.

**Sample Request**

```plaintext
POST / HTTP/1.1
Host: gamelift.us-west-2.amazonaws.com;
Accept-Encoding: identity
Content-Length: 1249
User-Agent: aws-cli/1.11.36 Python/2.7.9 Windows/7 botocore/1.4.93
Content-Type: application/x-amz-json-1.0
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20170406/us-west-2/gamelift/aws4_request, SignedHeaders=content-type;host;x-amz-date;x-amz-target,
Signature=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
X-Amz-Date: 20170406T004805Z
X-Amz-Target: GameLift.CreateFleet

{
  "Name": "My_Fleet_1",
  "Description": "A fully configured sample fleet",
  "BuildId": "build-1111aaaa-22bb-33cc-44dd-5555eeee66ff",
  "EC2InstanceType": "c4.large",
  "EC2InboundPermissions": [
    {
      "FromPort": 33435,
      "ToPort": 33435,
      "IpRange": "10.24.34.0/23",
      "Protocol": "UDP"
    },
    {
      "FromPort": 33235,
      "ToPort": 33235,
      "IpRange": "10.24.34.0/23",
      "Protocol": "UDP"
    }
  ],
  "FleetType": "ON_DEMAND",
  "NewGameSessionProtectionPolicy": "FullProtection",
  "RuntimeConfiguration": {
    "ServerProcesses": [
      {
        "LaunchPath": "c:\game\mygame.exe",
        "Parameters": "+map Winter444",
        "ConcurrentExecutions": 3
      },
      {
        "LaunchPath": "c:\game\mygame.exe",
        "Parameters": "-dev -console -sw +sv_lan 1 +sv_cheats 1 +port 33235 +map Winter444",
        "ConcurrentExecutions": 1
      }
    ],
    "MaxConcurrentGameSessionActivations": 2,
    "GameSessionActivationTimeoutSeconds": 300
  },
  "ResourceCreationLimitPolicy": {
    "NewGameSessionsPerCreator": 3,
    "PolicyPeriodInMinutes": 15
  },
  "MetricGroups": ["EMEAfleets"],
  "PeerVpcAwsAccountId": "444455556666",
  "PeerVpcId": "vpc-a11a11a1"
}

CLI syntax:

```
aws gamelift create-fleet
  --name "My_Fleet_2"
```

API Version 2015-10-01
28
---description "Another fully configured sample fleet"
--build-id "build-1111aaaa-22bb-33cc-44dd-5555eee66ff"
--ec2-instance-type "c4.large"
--ec2-inbound-permissions
"FromPort=33435,ToPort=33435,IpRange=10.24.34.0/23,Protocol=UDP"
--fleet-type "ON-DEMAND"
--new-game-session-protection-policy "FullProtection"
--runtime-configuration
"GameSessionActivationTimeoutSeconds=300,
MaxConcurrentGameSessionActivations=2,
ServerProcesses=[
    {LaunchPath=c:\game\mygame.exe,Parameters=+map Winter444,ConcurrentExecutions=1},
    {LaunchPath=c:\game\mygame.exe,Parameters=+dev -console -sw +sv_lan 1 +sv_cheats 1
+port 33235 +map Winter444,ConcurrentExecutions=1}
]"
--resource-creation-limit-policy "NewGameSessionsPerCreator=3,PolicyPeriodInMinutes=15"
--metric-groups "EMEAfleets"
--peer-vpc-aws-account-id "401680102694"
--peer-vpc-id "vpc-f12bf994"

Sample Response

HTTP/1.1 200 OK
x-amzn-RequestId: b34f8665-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 699
Date: Thu, 06 Apr 2017 00:48:07 GMT

{
    "FleetAttributes": {
        "BuildId": "build-1111aaaa-22bb-33cc-44dd-5555eee66ff",
        "CreationTime": 1496375088.502,
        "Description": "A fully configured sample fleet",
        "FleetArn": "arn:aws:gamelift:us-west-2::fleet/
        fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
        "FleetId": "fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
        "FleetType": "ON_DEMAND",
        "MetricGroups": ["EMEAfleets"],
        "Name": "My_Fleet_One",
        "NewGameSessionProtectionPolicy": "FullProtection",
        "OperatingSystem": "WINDOWS_2012",
        "ResourceCreationLimitPolicy": {
            "NewGameSessionsPerCreator": 3,
            "PolicyPeriodInMinutes": 15
        }
    }
}

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
See Also

- AWS SDK for Python
- AWS SDK for Ruby V2
CreateGameSession

Creates a multiplayer game session for players. This action creates a game session record and assigns an available server process in the specified fleet to host the game session. A fleet must have an `ACTIVE` status before a game session can be created in it.

To create a game session, specify either fleet ID or alias ID and indicate a maximum number of players to allow in the game session. You can also provide a name and game-specific properties for this game session. If successful, a `GameSession` object is returned containing the game session properties and other settings you specified.

**Idempotency tokens.** You can add a token that uniquely identifies game session requests. This is useful for ensuring that game session requests are idempotent. Multiple requests with the same idempotency token are processed only once; subsequent requests return the original result. All response values are the same with the exception of game session status, which may change.

**Resource creation limits.** If you are creating a game session on a fleet with a resource creation limit policy in force, then you must specify a creator ID. Without this ID, Amazon GameLift has no way to evaluate the policy for this new game session request.

**Player acceptance policy.** By default, newly created game sessions are open to new players. You can restrict new player access by using `UpdateGameSession` to change the game session's player session creation policy.

**Game session logs.** Logs are retained for all active game sessions for 14 days. To access the logs, call `GetGameSessionLogUrl` to download the log files.

*Available in Amazon GameLift Local.*

Game-session-related operations include:

- `CreateGameSession`
- `DescribeGameSessions`
- `DescribeGameSessionDetails`
- `SearchGameSessions`
- `UpdateGameSession`
- `GetGameSessionLogUrl`
- `Game session placements`
  - `StartGameSessionPlacement`
  - `DescribeGameSessionPlacement`
  - `StopGameSessionPlacement`

**Request Syntax**

```json
{
    "AliasId": "string",
    "CreatorId": "string",
    "FleetId": "string",
    "GameProperties": [
        {
            "Key": "string",
            "Value": "string"
        }
    ],
    "GameSessionData": "string",
}
```

API Version 2015-10-01
Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

Note
In the following list, the required parameters are described first.

**MaximumPlayerSessionCount (p. 31)**

Maximum number of players that can be connected simultaneously to the game session.

Type: Integer

Valid Range: Minimum value of 0.

Required: Yes

**AliasId (p. 31)**

Unique identifier for an alias associated with the fleet to create a game session in. Each request must reference either a fleet ID or alias ID, but not both.

Type: String

Pattern: ^alias-\S+

Required: No

**CreatorId (p. 31)**

Unique identifier for a player or entity creating the game session. This ID is used to enforce a resource protection policy (if one exists) that limits the number of concurrent active game sessions one player can have.

Type: String


Required: No

**FleetId (p. 31)**

Unique identifier for a fleet to create a game session in. Each request must reference either a fleet ID or alias ID, but not both.

Type: String

Pattern: ^fleet-\S+

Required: No

**GameProperties (p. 31)**

Set of custom properties for a game session, formatted as key:value pairs. These properties are passed to a game server process in the GameSession (p. 290) object with a request to start a new game session (see Start a Game Session).
Type: Array of GameProperty (p. 289) objects

Array Members: Maximum number of 16 items.

Required: No

GameSessionData (p. 31)

Set of custom game session properties, formatted as a single string value. This data is passed to a game server process in the GameSession (p. 290) object with a request to start a new game session (see Start a Game Session).

Type: String


Required: No

GameSessionId (p. 31)

This parameter is no longer preferred. Please use IdempotencyToken instead. Custom string that uniquely identifies a request for a new game session. Maximum token length is 48 characters. If provided, this string is included in the new game session's ID. (A game session ARN has the following format: arn:aws:gamelift:<region>::gamesession/<fleet ID>/<custom ID string or idempotency token>.)

Type: String


Pattern: [a-zA-Z0-9-]+

Required: No

IdempotencyToken (p. 31)

Custom string that uniquely identifies a request for a new game session. Maximum token length is 48 characters. If provided, this string is included in the new game session's ID. (A game session ARN has the following format: arn:aws:gamelift:<region>::gamesession/<fleet ID>/<custom ID string or idempotency token>.) Idempotency tokens remain in use for 30 days after a game session has ended; game session objects are retained for this time period and then deleted.

Type: String


Pattern: [a-zA-Z0-9-]+

Required: No

Name (p. 31)

Descriptive label that is associated with a game session. Session names do not need to be unique.

Type: String


Required: No

Response Syntax

```json
{
```
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**GameSession (p. 33)**

Object that describes the newly created game session record.

Type: GameSession (p. 290) object

**Errors**

For information about the errors that are common to all actions, see Common Errors (p. 347).

**ConflictException**

The requested operation would cause a conflict with the current state of a service resource associated with the request. Resolve the conflict before retrying this request.

HTTP Status Code: 400

**FleetCapacityExceededException**

The specified fleet has no available instances to fulfill a CreateGameSession request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 400

**IdempotentParameterMismatchException**

A game session with this custom ID string already exists in this fleet. Resolve this conflict before retrying this request.

HTTP Status Code: 400
InternalServiceException

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

InvalidFleetStatusException

The requested operation would cause a conflict with the current state of a resource associated with the request and/or the fleet. Resolve the conflict before retrying.

HTTP Status Code: 400

InvalidRequestException

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

LimitExceededException

The requested operation would cause the resource to exceed the allowed service limit. Resolve the issue before retrying.

HTTP Status Code: 400

NotFoundException

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

TerminalRoutingStrategyException

The service is unable to resolve the routing for a particular alias because it has a terminal RoutingStrategy (p. 329) associated with it. The message returned in this exception is the message defined in the routing strategy itself. Such requests should only be retried if the routing strategy for the specified alias is modified.

HTTP Status Code: 400

UnauthorizedException

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V2
CreateGameSessionQueue

Establishes a new queue for processing requests to place new game sessions. A queue identifies where new game sessions can be hosted -- by specifying a list of destinations (fleets or aliases) -- and how long requests can wait in the queue before timing out. You can set up a queue to try to place game sessions on fleets in multiple regions. To add placement requests to a queue, call StartGameSessionPlacement (p. 204) and reference the queue name.

Destination order. When processing a request for a game session, Amazon GameLift tries each destination in order until it finds one with available resources to host the new game session. A queue's default order is determined by how destinations are listed. The default order is overridden when a game session placement request provides player latency information. Player latency information enables Amazon GameLift to prioritize destinations where players report the lowest average latency, as a result placing the new game session where the majority of players will have the best possible gameplay experience.

Player latency policies. For placement requests containing player latency information, use player latency policies to protect individual players from very high latencies. With a latency cap, even when a destination can deliver a low latency for most players, the game is not placed where any individual player is reporting latency higher than a policy's maximum. A queue can have multiple latency policies, which are enforced consecutively starting with the policy with the lowest latency cap. Use multiple policies to gradually relax latency controls; for example, you might set a policy with a low latency cap for the first 60 seconds, a second policy with a higher cap for the next 60 seconds, etc.

To create a new queue, provide a name, timeout value, a list of destinations and, if desired, a set of latency policies. If successful, a new queue object is returned.

Queue-related operations include:

- CreateGameSessionQueue (p. 37)
- DescribeGameSessionQueues (p. 120)
- UpdateGameSessionQueue (p. 250)
- DeleteGameSessionQueue (p. 73)

Request Syntax

```json
{
  "Destinations": [
    {
      "DestinationArn": "string"
    }
  ],
  "Name": "string",
  "PlayerLatencyPolicies": [
    {
      "MaximumIndividualPlayerLatencyMilliseconds": number,
      "PolicyDurationSeconds": number
    }
  ],
  "TimeoutInSeconds": number
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).
The request accepts the following data in JSON format.

**Note**
In the following list, the required parameters are described first.

**Name (p. 37)**
Descriptive label that is associated with game session queue. Queue names must be unique within each region.

Type: String
Pattern: [a-zA-Z0-9-]+
Required: Yes

**Destinations (p. 37)**
List of fleets that can be used to fulfill game session placement requests in the queue. Fleets are identified by either a fleet ARN or a fleet alias ARN. Destinations are listed in default preference order.

Type: Array of `GameSessionQueueDestination (p. 303)` objects
Required: No

**PlayerLatencyPolicies (p. 37)**
Collection of latency policies to apply when processing game sessions placement requests with player latency information. Multiple policies are evaluated in order of the maximum latency value, starting with the lowest latency values. With just one policy, it is enforced at the start of the game session placement for the duration period. With multiple policies, each policy is enforced consecutively for its duration period. For example, a queue might enforce a 60-second policy followed by a 120-second policy, and then no policy for the remainder of the placement. A player latency policy must set a value for `MaximumIndividualPlayerLatencyMilliseconds`; if none is set, this API requests will fail.

Type: Array of `PlayerLatencyPolicy (p. 324)` objects
Required: No

**TimeoutInSeconds (p. 37)**
Maximum time, in seconds, that a new game session placement request remains in the queue. When a request exceeds this time, the game session placement changes to a `TIMED_OUT` status.

Type: Integer
Valid Range: Minimum value of 0.
Required: No

### Response Syntax

```json
{
    "GameSessionQueue": {
        "Destinations": [
            {
                "DestinationArn": "string"
            }
        ],
        "GameSessionQueueArn": "string",
```
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

GameSessionQueue (p. 38)
Object that describes the newly created game session queue.

Type: GameSessionQueue (p. 301) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

InternalServiceException
The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

InvalidRequestException
One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

LimitExceededException
The requested operation would cause the resource to exceed the allowed service limit. Resolve the issue before retrying.

HTTP Status Code: 400

UnauthorizedException
The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

Example

Create and configure a game session queue

This example creates a new game session queue with destinations in two different regions and a timeout of 10 minutes. It configures two latency policies, which are enforced consecutively starting with the
one with the lowest maximum latency value. In this example, the 100ms latency cap is enforced during the first 60 seconds of a game session placement, followed by the 200ms cap for rest of the placement duration.

HTTP requests are authenticated using an AWS Signature Version 4 signature in the Authorization header field.

**Sample Request**

```
POST / HTTP/1.1
Host: gamelift.us-west-2.amazonaws.com;
Accept-Encoding: identity
Content-Length: 508
User-Agent: aws-cli/1.11.36 Python/2.7.9 Windows/7 botocore/1.4.93
Content-Type: application/x-amz-json-1.0
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20170406/us-west-2/gamelift/aws4_request, SignedHeaders=content-type;host;x-amz-date;x-amz-target, Signature=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
X-Amz-Date: 20170406T004805Z
X-Amz-Target: GameLift.CreateGameSessionQueue
{
    "Destinations": [
        { "DestinationArn": "arn:aws:gamelift:us-west-2::fleet/fleet-1a2b3c4d-56e7-7a8b-9c0d-1e2f3a4b56c6d" },
        { "DestinationArn": "arn:aws:gamelift:us-east-1::fleet/fleet-5c6d3c4d-56e7-7a8b-9c0d-1e2f3a4b5a2a" }]
    
    "Name": "matchmaker-queue",
    "PlayerLatencyPolicies": [
        { "MaximumIndividualPlayerLatencyMilliseconds": 200 },
        { "MaximumIndividualPlayerLatencyMilliseconds": 100, "PolicyDurationSeconds": 60 }
    ],
    "TimeoutInSeconds": 600
}
```

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
CreateMatchmakingConfiguration

Defines a new matchmaking configuration for use with FlexMatch. A matchmaking configuration sets out guidelines for matching players and getting the matches into games. You can set up multiple matchmaking configurations to handle the scenarios needed for your game. Each matchmaking ticket (StartMatchmaking (p. 214) or StartMatchBackfill (p. 210)) specifies a configuration for the match and provides player attributes to support the configuration being used.

To create a matchmaking configuration, at a minimum you must specify the following: configuration name; a rule set that governs how to evaluate players and find acceptable matches; a game session queue to use when placing a new game session for the match; and the maximum time allowed for a matchmaking attempt.

**Player acceptance** -- In each configuration, you have the option to require that all players accept participation in a proposed match. To enable this feature, set `AcceptanceRequired` to true and specify a time limit for player acceptance. Players have the option to accept or reject a proposed match, and a match does not move ahead to game session placement unless all matched players accept.

**Matchmaking status notification** -- There are two ways to track the progress of matchmaking tickets: (1) polling ticket status with DescribeMatchmaking (p. 131); or (2) receiving notifications with Amazon Simple Notification Service (SNS). To use notifications, you first need to set up an SNS topic to receive the notifications, and provide the topic ARN in the matchmaking configuration (see Setting up Notifications for Matchmaking). Since notifications promise only "best effort" delivery, we recommend calling DescribeMatchmaking if no notifications are received within 30 seconds.

Operations related to match configurations and rule sets include:

- CreateMatchmakingConfiguration (p. 41)
- DescribeMatchmakingConfigurations (p. 134)
- UpdateMatchmakingConfiguration (p. 253)
- DeleteMatchmakingConfiguration (p. 75)
- CreateMatchmakingRuleSet (p. 46)
- DescribeMatchmakingRuleSets (p. 137)
- ValidateMatchmakingRuleSet (p. 262)

**Request Syntax**

```json
{
    "AcceptanceRequired": boolean,
    "AcceptanceTimeoutSeconds": number,
    "AdditionalPlayerCount": number,
    "CustomEventData": "string",
    "Description": "string",
    "GameProperties": [
        {
            "Key": "string",
            "Value": "string"
        }
    ],
    "GameSessionData": "string",
    "GameSessionQueueArns": [ "string" ],
    "Name": "string",
    "NotificationTarget": "string",
    "RequestTimeoutSeconds": number,
    "RuleSetName": "string"
}
```

API Version 2015-10-01
Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**
In the following list, the required parameters are described first.

**AcceptanceRequired (p. 41)**

Flag that determines whether or not a match that was created with this configuration must be accepted by the matched players. To require acceptance, set to TRUE.

Type: Boolean

Required: Yes

**GameSessionQueueArns (p. 41)**

Amazon Resource Name (ARN) that is assigned to a game session queue and uniquely identifies it. Format is `arn:aws:gamelift:<region>::fleet/fleet-a1234567-b8c9-0d1e-2fa3-b45c6d7e8912`. These queues are used when placing game sessions for matches that are created with this matchmaking configuration. Queues can be located in any region.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: `[a-zA-Z0-9:/-]+`

Required: Yes

**Name (p. 41)**

Unique identifier for a matchmaking configuration. This name is used to identify the configuration associated with a matchmaking request or ticket.

Type: String


Pattern: `[a-zA-Z0-9-\.]+`

Required: Yes

**RequestTimeoutSeconds (p. 41)**

Maximum duration, in seconds, that a matchmaking ticket can remain in process before timing out. Requests that time out can be resubmitted as needed.

Type: Integer


Required: Yes

**RuleSetName (p. 41)**

Unique identifier for a matchmaking rule set to use with this configuration. A matchmaking configuration can only use rule sets that are defined in the same region.
Type: String


Pattern: \[a-zA-Z0-9-\.]+\]

Required: Yes

**AcceptanceTimeoutSeconds (p. 41)**

Length of time (in seconds) to wait for players to accept a proposed match. If any player rejects the match or fails to accept before the timeout, the ticket continues to look for an acceptable match.

Type: Integer


Required: No

**AdditionalPlayerCount (p. 41)**

Number of player slots in a match to keep open for future players. For example, if the configuration's rule set specifies a match for a single 12-person team, and the additional player count is set to 2, only 10 players are selected for the match.

Type: Integer

Valid Range: Minimum value of 0.

Required: No

**CustomEventData (p. 41)**

Information to attached to all events related to the matchmaking configuration.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 256.

Required: No

**Description (p. 41)**

Meaningful description of the matchmaking configuration.

Type: String


Required: No

**GameProperties (p. 41)**

Set of custom properties for a game session, formatted as key:value pairs. These properties are passed to a game server process in the GameSession (p. 290) object with a request to start a new game session (see Start a Game Session). This information is added to the new GameSession (p. 290) object that is created for a successful match.

Type: Array of GameProperty (p. 289) objects

Array Members: Maximum number of 16 items.

Required: No
GameSessionData (p. 41)

Set of custom game session properties, formatted as a single string value. This data is passed to a
game server process in the GameSession (p. 290) object with a request to start a new game session
(see Start a Game Session). This information is added to the new GameSession (p. 290) object that
is created for a successful match.

Type: String


Required: No

NotificationTarget (p. 41)

SNS topic ARN that is set up to receive matchmaking notifications.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 300.

Pattern: [a-zA-Z0-9_:/-]*

Required: No

Response Syntax

```json
{
  "Configuration": {
    "AcceptanceRequired": boolean,
    "AcceptanceTimeoutSeconds": number,
    "AdditionalPlayerCount": number,
    "CreationTime": number,
    "CustomEventData": "string",
    "Description": "string",
    "GameProperties": [
      {
        "Key": "string",
        "Value": "string"
      }
    ],
    "GameSessionData": "string",
    "GameSessionQueueArns": [ "string" ],
    "Name": "string",
    "NotificationTarget": "string",
    "RequestTimeoutSeconds": number,
    "RuleSetName": "string"
  }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

Configuration (p. 44)

Object that describes the newly created matchmaking configuration.
Type: MatchmakingConfiguration (p. 312) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

InternalServiceException

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

InvalidRequestException

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

LimitExceededException

The requested operation would cause the resource to exceed the allowed service limit. Resolve the issue before retrying.

HTTP Status Code: 400

NotFoundException

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

UnsupportedRegionException

The requested operation is not supported in the region specified.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
CreateMatchmakingRuleSet

Creates a new rule set for FlexMatch matchmaking. A rule set describes the type of match to create, such as the number and size of teams, and sets the parameters for acceptable player matches, such as minimum skill level or character type. Rule sets are used in matchmaking configurations, which define how matchmaking requests are handled. Each MatchmakingConfiguration (p. 312) uses one rule set; you can set up multiple rule sets to handle the scenarios that suit your game (such as for different game modes), and create a separate matchmaking configuration for each rule set. See additional information on rule set content in the MatchmakingRuleSet (p. 315) structure. For help creating rule sets, including useful examples, see the topic Adding FlexMatch to Your Game.

Once created, matchmaking rule sets cannot be changed or deleted, so we recommend checking the rule set syntax using ValidateMatchmakingRuleSet (p. 262) before creating the rule set.

To create a matchmaking rule set, provide the set of rules and a unique name. Rule sets must be defined in the same region as the matchmaking configuration they will be used with. Rule sets cannot be edited or deleted. If you need to change a rule set, create a new one with the necessary edits and then update matchmaking configurations to use the new rule set.

Operations related to match configurations and rule sets include:

- CreateMatchmakingConfiguration (p. 41)
- DescribeMatchmakingConfigurations (p. 134)
- UpdateMatchmakingConfiguration (p. 253)
- DeleteMatchmakingConfiguration (p. 75)
- CreateMatchmakingRuleSet (p. 46)
- DescribeMatchmakingRuleSets (p. 137)
- ValidateMatchmakingRuleSet (p. 262)

Request Syntax

```json
{
   "Name": "string",
   "RuleSetBody": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.

**Name (p. 46)**

Unique identifier for a matchmaking rule set. This name is used to identify the rule set associated with a matchmaking configuration.

Type: String

Pattern: [a-zA-Z0-9-\.]+  
Required: Yes

**RuleSetBody (p. 46)**

Collection of matchmaking rules, formatted as a JSON string. (Note that comments are not allowed in JSON, but most elements support a description field.)

Type: String


Required: Yes

**Response Syntax**

```
{
   "RuleSet": {
      "CreationTime": number,
      "RuleSetBody": "string",
      "RuleSetName": "string"
   }
}
```

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**RuleSet (p. 47)**

Object that describes the newly created matchmaking rule set.

Type: **MatchmakingRuleSet (p. 315)** object

**Errors**

For information about the errors that are common to all actions, see Common Errors (p. 347).

**InternalServiceException**

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

**InvalidRequestException**

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

**UnsupportedRegionException**

The requested operation is not supported in the region specified.
HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
CreatePlayerSession

Adds a player to a game session and creates a player session record. Before a player can be added, a game session must have an ACTIVE status, have a creation policy of ALLOW_ALL, and have an open player slot. To add a group of players to a game session, use CreatePlayerSessions (p. 52).

To create a player session, specify a game session ID, player ID, and optionally a string of player data. If successful, the player is added to the game session and a new PlayerSession (p. 325) object is returned. Player sessions cannot be updated.

Available in Amazon GameLift Local.

Player-session-related operations include:

- CreatePlayerSession (p. 49)
- CreatePlayerSessions (p. 52)
- DescribePlayerSessions (p. 140)
- Game session placements
  - StartGameSessionPlacement (p. 204)
  - DescribeGameSessionPlacement (p. 117)
  - StopGameSessionPlacement (p. 222)

Request Syntax

```json
{
  "GameSessionId": "string",
  "PlayerData": "string",
  "PlayerId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.

**GameSessionId** (p. 49)

Unique identifier for the game session to add a player to.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: `[^a-zA-Z0-9:/-]+`

Required: Yes

**PlayerId** (p. 49)

Unique identifier for a player. Player IDs are developer-defined.

Type: String
Response Syntax

```
{
   "PlayerSession": {
      "CreationTime": number,
      "FleetId": "string",
      "GameSessionId": "string",
      "IpAddress": "string",
      "PlayerData": "string",
      "PlayerId": "string",
      "PlayerSessionId": "string",
      "Port": number,
      "Status": "string",
      "TerminationTime": number
   }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**PlayerSession (p. 50)**

Object that describes the newly created player session record.

Type: PlayerSession (p. 325) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

**GameSessionFullException**

The game instance is currently full and cannot allow the requested player(s) to join. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 400

**InternalServiceException**

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.
HTTP Status Code: 500
InvalidGameSessionStatusException

The requested operation would cause a conflict with the current state of a resource associated with the request and/or the game instance. Resolve the conflict before retrying.

HTTP Status Code: 400
InvalidRequestException

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400
NotFoundException

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400
TerminalRoutingStrategyException

The service is unable to resolve the routing for a particular alias because it has a terminal RoutingStrategy (p. 329) associated with it. The message returned in this exception is the message defined in the routing strategy itself. Such requests should only be retried if the routing strategy for the specified alias is modified.

HTTP Status Code: 400
UnauthorizedException

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
CreatePlayerSessions

Adds a group of players to a game session. This action is useful with a team matching feature. Before players can be added, a game session must have an ACTIVE status, have a creation policy of ALLOW_ALL, and have an open player slot. To add a single player to a game session, use CreatePlayerSession (p. 49).

To create player sessions, specify a game session ID, a list of player IDs, and optionally a set of player data strings. If successful, the players are added to the game session and a set of new PlayerSession (p. 325) objects is returned. Player sessions cannot be updated.

Available in Amazon GameLift Local.

Player-session-related operations include:
- CreatePlayerSession (p. 49)
- CreatePlayerSessions (p. 52)
- DescribePlayerSessions (p. 140)
- Game session placements
  - StartGameSessionPlacement (p. 204)
  - DescribeGameSessionPlacement (p. 117)
  - StopGameSessionPlacement (p. 222)

Request Syntax

```json
{
    "GameSessionId": "string",
    "PlayerDataMap": {
        "string": "string"
    },
    "PlayerIds": [ "string" ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.

**GameSessionId (p. 52)**

Unique identifier for the game session to add players to.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: `[a-zA-Z0-9:-]+`

Required: Yes

**PlayerIds (p. 52)**

List of unique identifiers for the players to be added.
Response Syntax

```json
{
    "PlayerSessions": [
        {
            "CreationTime": number,
            "FleetId": "string",
            "GameSessionId": "string",
            "IpAddress": "string",
            "PlayerData": "string",
            "PlayerId": "string",
            "PlayerSessionId": "string",
            "Port": number,
            "Status": "string",
            "TerminationTime": number
        }
    ]
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**PlayerSessions (p. 53)**

Collection of player session objects created for the added players.

Type: Array of PlayerSession (p. 325) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).
**GameSessionFullException**

The game instance is currently full and cannot allow the requested player(s) to join. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 400

**InternalServiceException**

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

**InvalidGameSessionStatusException**

The requested operation would cause a conflict with the current state of a resource associated with the request and/or the game instance. Resolve the conflict before retrying.

HTTP Status Code: 400

**InvalidRequestException**

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

**NotFoundException**

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

**TerminalRoutingStrategyException**

The service is unable to resolve the routing for a particular alias because it has a terminal RoutingStrategy (p. 329) associated with it. The message returned in this exception is the message defined in the routing strategy itself. Such requests should only be retried if the routing strategy for the specified alias is modified.

HTTP Status Code: 400

**UnauthorizedException**

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V2
CreateVpcPeeringAuthorization

Requests authorization to create or delete a peer connection between the VPC for your Amazon GameLift fleet and a virtual private cloud (VPC) in your AWS account. VPC peering enables the game servers on your fleet to communicate directly with other AWS resources. Once you've received authorization, call CreateVpcPeeringConnection (p. 61) to establish the peering connection. For more information, see VPC Peering with Amazon GameLift Fleets.

You can peer with VPCs that are owned by any AWS account you have access to, including the account that you use to manage your Amazon GameLift fleets. You cannot peer with VPCs that are in different regions.

To request authorization to create a connection, call this operation from the AWS account with the VPC that you want to peer to your Amazon GameLift fleet. For example, to enable your game servers to retrieve data from a DynamoDB table, use the account that manages that DynamoDB resource. Identify the following values: (1) The ID of the VPC that you want to peer with, and (2) the ID of the AWS account that you use to manage Amazon GameLift. If successful, VPC peering is authorized for the specified VPC.

To request authorization to delete a connection, call this operation from the AWS account with the VPC that is peered with your Amazon GameLift fleet. Identify the following values: (1) VPC ID that you want to delete the peering connection for, and (2) ID of the AWS account that you use to manage Amazon GameLift.

The authorization remains valid for 24 hours unless it is canceled by a call to DeleteVpcPeeringAuthorization (p. 80). You must create or delete the peering connection while the authorization is valid.

VPC peering connection operations include:

- CreateVpcPeeringAuthorization (p. 56)
- DescribeVpcPeeringAuthorizations (p. 152)
- DeleteVpcPeeringAuthorization (p. 80)
- CreateVpcPeeringConnection (p. 61)
- DescribeVpcPeeringConnections (p. 154)
- DeleteVpcPeeringConnection (p. 82)

Request Syntax

```json
{
    "GameLiftAwsAccountId": "string",
    "PeerVpcId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.
**GameLiftAwsAccountId (p. 56)**

Unique identifier for the AWS account that you use to manage your Amazon GameLift fleet. You can find your Account ID in the AWS Management Console under account settings.

Type: String


Required: Yes

**PeerVpcId (p. 56)**

Unique identifier for a VPC with resources to be accessed by your Amazon GameLift fleet. The VPC must be in the same region where your fleet is deployed. To get VPC information, including IDs, use the Virtual Private Cloud service tools, including the VPC Dashboard in the AWS Management Console.

Type: String


Required: Yes

**Response Syntax**

```
{
    "VpcPeeringAuthorization": {
        "CreationTime": number,
        "ExpirationTime": number,
        "GameLiftAwsAccountId": "string",
        "PeerVpcAwsAccountId": "string",
        "PeerVpcId": "string"
    }
}
```

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**VpcPeeringAuthorization (p. 57)**

Details on the requested VPC peering authorization, including expiration.

Type: VpcPeeringAuthorization (p. 340) object

**Errors**

For information about the errors that are common to all actions, see Common Errors (p. 347).

**InternalServiceException**

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500
InvalidRequestException

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

NotFoundException

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

UnauthorizedException

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

Examples

Authorize VPC peering between your Amazon GameLift fleet and resources on your Amazon GameLift AWS account

In this example, you want your game servers that are running on an Amazon GameLift fleet to be able to access a web service. The web service is managed through the same AWS account that you use to manage your Amazon GameLift fleet (account ID is 111122223333). You've already created a VPC (or you're using your account's default VPC) for the web service. The ID for this VPC is vpc-a12bc345.

To make this request, sign in using your credentials for AWS account 111122223333.

HTTP requests are authenticated using an AWS Signature Version 4 signature in the Authorization header field.

Sample Request

POST / HTTP/1.1
Host: gamelift.us-west-2.amazonaws.com;
Accept-Encoding: identity
Content-Length: 77
User-Agent: aws-cli/1.11.36 Python/2.7.9 Windows/7 botocore/1.4.93
Content-Type: application/x-amz-json-1.0
Authorization: AWS4-HMAC-SHA256  Credential=AKIAIOSFODNN7EXAMPLE/20170406/us-west-2/gamelift/aws4_request, SignedHeaders=content-type;host;x-amz-date;x-amz-target,
   Signature=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
X-Amz-Date: 20170406T004805Z
X-Amz-Target: GameLift.CreateVpcPeeringAuthorization

{ "GameLiftAwsAccountId": "111122223333",
  "PeerVpcId": "vpc-a12bc345"
}

Sample Response

HTTP/1.1 200 OK
x-amzn-RequestId: b34f8665-EXAMPLE
Content-Type: application/x-amz-json-1.1
Authorize VPC peering between your Amazon GameLift fleet and resources on a different AWS account

As in the previous example, you want your game servers to be able to access a web service. But in this example, the web service is managed through a different account from the one that you use to manage your Amazon GameLift fleet. Your Amazon GameLift account ID is 111122223333, while the web service account ID is 444455556666. A VPC has already been created on account 444455556666 with the web service. The ID for this VPC is vpc-c67ef890.

To make this request, sign in using credentials for AWS account 444455556666. If you don't have rights to this account, you need to provide your Amazon GameLift account ID to the owner of AWS account 444455556666 to make the request.

HTTP requests are authenticated using an AWS Signature Version 4 signature in the Authorization header field.

Sample Request

```
POST / HTTP/1.1
Host: gamelift.us-west-2.amazonaws.com;
Accept-Encoding: identity
Content-Length: 82
User-Agent: aws-cli/1.11.36 Python/2.7.9 Windows/7 botocore/1.4.93
Content-Type: application/x-amz-json-1.0
Authorization: AWS4-HMAC-SHA256  Credential=AKIAIOSFODNN7EXAMPLE/20170406/us-west-2/gamelift/aws4_request, SignedHeaders=content-type;host;x-amz-date;x-amz-target,
Signature=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
X-Amz-Date: 20170406T004805Z
X-Amz-Target: GameLift.CreateVpcPeeringAuthorization
{
    "GameLiftAwsAccountId": "111122223333",
    "PeerVpcId": "vpc-c67ef890"
}
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: b34f8665-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 225
Date: Thu, 06 Apr 2017 00:48:07 GMT

{"VpcPeeringAuthorization":
    {"CreationTime": 1503608847.489,
     "ExpirationTime": 1503695247,
     "GameLiftAwsAccountId": "111122223333",
     "PeerVpcAwsAccountId": "111122223333",
     "PeerVpcId": "vpc-c67ef890"}
}
```
See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
CreateVpcPeeringConnection

Establishes a VPC peering connection between a virtual private cloud (VPC) in an AWS account with the VPC for your Amazon GameLift fleet. VPC peering enables the game servers on your fleet to communicate directly with other AWS resources. You can peer with VPCs in any AWS account that you have access to, including the account that you use to manage your Amazon GameLift fleets. You cannot peer with VPCs that are in different regions. For more information, see VPC Peering with Amazon GameLift Fleets.

Before calling this operation to establish the peering connection, you first need to call CreateVpcPeeringAuthorization (p. 56) and identify the VPC you want to peer with. Once the authorization for the specified VPC is issued, you have 24 hours to establish the connection. These two operations handle all tasks necessary to peer the two VPCs, including acceptance, updating routing tables, etc.

To establish the connection, call this operation from the AWS account that is used to manage the Amazon GameLift fleets. Identify the following values: (1) The ID of the fleet you want to enable a VPC peering connection for; (2) The AWS account with the VPC that you want to peer with; and (3) The ID of the VPC you want to peer with. This operation is asynchronous. If successful, a VpcPeeringConnection (p. 342) request is created. You can use continuous polling to track the request's status using DescribeVpcPeeringConnections (p. 154), or by monitoring fleet events for success or failure using DescribeFleetEvents (p. 101).

VPC peering connection operations include:

- CreateVpcPeeringAuthorization (p. 56)
- DescribeVpcPeeringAuthorizations (p. 152)
- DeleteVpcPeeringAuthorization (p. 80)
- CreateVpcPeeringConnection (p. 61)
- DescribeVpcPeeringConnections (p. 154)
- DeleteVpcPeeringConnection (p. 82)

**Request Syntax**

```
{
  "FleetId": "string",
  "PeerVpcAwsAccountId": "string",
  "PeerVpcId": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.

**FleetId (p. 61)**

Unique identifier for a fleet. This tells Amazon GameLift which GameLift VPC to peer with.

Type: String
Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

**InternalServiceException**

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

**InvalidRequestException**

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

**NotFoundException**

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

**UnauthorizedException**

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400
Examples

Peer the VPC for your Amazon GameLift fleet with a VPC on your Amazon GameLift AWS account

This example builds on Example 1 in CreateVpcPeeringAuthorization. If authorization succeeded, the next step is to tell Amazon GameLift to request the peering connection. In this example, you want your game servers that are running on an Amazon GameLift fleet to be able to access a web service. The web service is managed through the same AWS account that you use to manage your Amazon GameLift fleet. To request the peering, provide the following details for the two VPCs to peer: (1) the Amazon GameLift fleet ID, and (2) the account and VPC for the web service. The account ID and VPC for the web service must be the same one you used in the authorization.

The fleet's ID is fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa. The AWS account (the same as the one you use to manage your Amazon GameLift fleet) is 111122223333. The VPC ID for the web service is vpc-a12bc345.

To make this request, sign in using your credentials for AWS account 111122223333. To view the resulting new connection record, call DescribeVpcPeeringConnections (p. 154) with the fleet ID.

HTTP requests are authenticated using an AWS Signature Version 4 signature in the Authorization header field.

Sample Request

POST / HTTP/1.1
Host: gamelift.us-west-2.amazonaws.com;
Accept-Encoding: identity
Content-Length: 141
User-Agent: aws-cli/1.11.36 Python/2.7.9 Windows/7 botocore/1.4.93
Content-Type: application/x-amz-json-1.0
Authorization: AWS4-HMAC-SHA256  Credential=AKIAIOSFODNN7EXAMPLE/20170406/us-west-2/gamelift/aws4_request, SignedHeaders=content-type;host;x-amz-date;x-amz-target,
Signature=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXMKfEYEY
X-Amz-Date: 20170406T004805Z
X-Amz-Target: GameLift.CreateVpcPeeringConnection

{
  "FleetId": "fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
  "PeerVpcAwsAccountId": "111122223333",
  "PeerVpcId": "vpc-a12bc345"
}

Peer the VPC for your Amazon GameLift fleet with a VPC on a different AWS account

This example builds on Example 2 in CreateVpcPeeringAuthorization. If authorization succeeded, the next step is to tell Amazon GameLift to request the peering connection. As in the previous example, you want your game servers that are running on an Amazon GameLift fleet to be able to access a web service. But in this example, the web service is managed by a different account from the one that you use to manage your Amazon GameLift fleet. To request the peering, provide the following details for the two VPCs to peer: (1) the Amazon GameLift fleet ID, and (2) the account and VPC for the web service. The account ID and VPC for the web service must be the same one you used in the authorization.

The fleet's ID is fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa. Your Amazon GameLift account ID is 111122223333. The AWS account with the web service is 444455556666. The VPC ID for the web service is vpc-c67ef890.
To make this request, sign in using your credentials for AWS account 111122223333 (your Amazon GameLift account). To view resulting new connection record, call `DescribeVpcPeeringConnections (p. 154)` with the fleet ID.

HTTP requests are authenticated using an AWS Signature Version 4 signature in the Authorization header field.

**Sample Request**

```plaintext
POST / HTTP/1.1
Host: gamelift.us-west-2.amazonaws.com;
Accept-Encoding: identity
Content-Length: 141
User-Agent: aws-cli/1.11.36 Python/2.7.9 Windows/7 botocore/1.4.93
Content-Type: application/x-amz-json-1.0
Authorization: AWS4-HMAC-SHA256  Credential=AKIAIOSFODNN7EXAMPLE/20170406/us-west-2/gamelift/aws4_request, SignedHeaders=content-type;host;x-amz-date;x-amz-target,
Signature=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
X-Amz-Date: 20170406T004805Z
X-Amz-Target: GameLift.CreateVpcPeeringConnection
{
  "FleetId": "fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
  "PeerVpcAwsAccountId": "444455556666",
  "PeerVpcId": "vpc-c67ef890"
}
```

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
DeleteAlias

Deletes an alias. This action removes all record of the alias. Game clients attempting to access a server process using the deleted alias receive an error. To delete an alias, specify the alias ID to be deleted.

Alias-related operations include:

- CreateAlias (p. 10)
- ListAliases (p. 165)
- DescribeAlias (p. 84)
- UpdateAlias (p. 227)
- DeleteAlias (p. 65)
- ResolveAlias (p. 189)

Request Syntax

```json
{
    "AliasId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.

**AliasId (p. 65)**

Unique identifier for a fleet alias. Specify the alias you want to delete.

- Type: String
- Pattern: ^alias-\S+
- Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

**InternalServiceException**

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.
HTTP Status Code: 500

**InvalidRequestException**

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

**NotFoundException**

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

**UnauthorizedException**

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
DeleteBuild

Deletes a build. This action permanently deletes the build record and any uploaded build files.

To delete a build, specify its ID. Deleting a build does not affect the status of any active fleets using the build, but you can no longer create new fleets with the deleted build.

Build-related operations include:

- CreateBuild (p. 13)
- ListBuilds (p. 168)
- DescribeBuild (p. 87)
- UpdateBuild (p. 230)
- DeleteBuild (p. 67)

Request Syntax

```json
{
    "BuildId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.

**BuildId (p. 67)**

Unique identifier for a build to delete.

Type: String

Pattern: ^build\-$

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

**InternalServiceException**

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.
Example

Remove a build

This example removes a build from your Amazon GameLift account. Once deleted, the build can no longer be used to create new fleets. This operation cannot be undone.

HTTP requests are authenticated using an AWS Signature Version 4 signature in the Authorization header field.

Sample Request

```plaintext
POST / HTTP/1.1
Host: gamelift.us-west-2.amazonaws.com;
Accept-Encoding: identity
Content-Length: 63
User-Agent: aws-cli/1.11.36 Python/2.7.9 Windows/7 botocore/1.4.93
Content-Type: application/x-amz-json-1.0
Authorization: AWS4-HMAC-SHA256  Credential=AKIAIOSFODNN7EXAMPLE/20170406/us-west-2/gamelift/aws4_request, SignedHeaders=content-type;host;x-amz-date;x-amz-target,
Signature=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
X-Amz-Date: 20170406T004805Z
X-Amz-Target: GameLift.DeleteBuild
{
    "BuildId": "build-1111aaaa-22bb-33cc-44dd-5555eeee66ff"
}
```

Sample Response

```plaintext
HTTP/1.1 200 OK undefined
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
See Also

- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
DeleteFleet

Deletes everything related to a fleet. Before deleting a fleet, you must set the fleet's desired capacity to zero. See UpdateFleetCapacity (p. 238).

This action removes the fleet's resources and the fleet record. Once a fleet is deleted, you can no longer use that fleet.

Fleet-related operations include:

- CreateFleet (p. 19)
- ListFleets (p. 173)
- DeleteFleet (p. 70)
- Describe fleets:
  - DescribeFleetAttributes (p. 93)
  - DescribeFleetCapacity (p. 97)
  - DescribeFleetPortSettings (p. 107)
  - DescribeFleetUtilization (p. 110)
  - DescribeRuntimeConfiguration (p. 144)
  - DescribeEC2InstanceLimits (p. 90)
  - DescribeFleetEvents (p. 101)
- Update fleets:
  - UpdateFleetAttributes (p. 234)
  - UpdateFleetCapacity (p. 238)
  - UpdateFleetPortSettings (p. 242)
  - UpdateRuntimeConfiguration (p. 258)
- Manage fleet actions:
  - StartFleetActions (p. 201)
  - StopFleetActions (p. 219)

Request Syntax

```json
{
   "FleetId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.

**FleetId (p. 70)**

Unique identifier for a fleet to be deleted.

Type: String
Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

**InternalServiceException**

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

**InvalidFleetStatusException**

The requested operation would cause a conflict with the current state of a resource associated with the request and/or the fleet. Resolve the conflict before retrying.

HTTP Status Code: 400

**InvalidRequestException**

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

**NotFoundException**

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

**UnauthorizedException**

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

Example

**Delete a fleet that is no longer in use**

This example attempts to delete a fleet that has been scaled down to zero instances. If the fleet capacity is not zero, the request fails with an HTTP 400 error.

HTTP requests are authenticated using an AWS Signature Version 4 signature in the Authorization header field.

**Sample Request**

```
POST / HTTP/1.1
```

API Version 2015-10-01
Sample Response

If delete is successful:

HTTP/1.1 200 OK
x-amzn-RequestId: b34f8665-EXAMPLE
Date: Thu, 06 Apr 2017 00:48:07 GMT

If delete is not successful:

Status: 400 Bad Request  RequestId: 99764159-4a57-11e7-af99-e5821c0e52da  Time: 177ms
{"__type": "com.amazonaws.gameliftapi.v20151001#InvalidRequestException", "Message": "Unable to terminate fleet-8e5a52a6-3261-4217-9a21-938b247c3db6: Max instance count must be 0."}

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
DeleteGameSessionQueue

Deletes a game session queue. This action means that any StartGameSessionPlacement (p. 204) requests that reference this queue will fail. To delete a queue, specify the queue name.

Queue-related operations include:

- CreateGameSessionQueue (p. 37)
- DescribeGameSessionQueues (p. 120)
- UpdateGameSessionQueue (p. 250)
- DeleteGameSessionQueue (p. 73)

Request Syntax

```json
{
  "Name": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.

**Name (p. 73)**

Descriptive label that is associated with game session queue. Queue names must be unique within each region.

Type: String


Pattern: [ a-zA-Z0-9- ]+

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

**InternalServiceException**

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.
HTTP Status Code: 500

InvalidRequestException

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

NotFoundException

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

UnauthorizedException

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
DeleteMatchmakingConfiguration

Permanently removes a FlexMatch matchmaking configuration. To delete, specify the configuration name. A matchmaking configuration cannot be deleted if it is being used in any active matchmaking tickets.

Operations related to match configurations and rule sets include:

- CreateMatchmakingConfiguration (p. 41)
- DescribeMatchmakingConfigurations (p. 134)
- UpdateMatchmakingConfiguration (p. 253)
- DeleteMatchmakingConfiguration (p. 75)
- CreateMatchmakingRuleSet (p. 46)
- DescribeMatchmakingRuleSets (p. 137)
- ValidateMatchmakingRuleSet (p. 262)

Request Syntax

```
{
  "Name": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.

**Name (p. 75)**

Unique identifier for a matchmaking configuration

Type: String


Pattern: `[a-zA-Z0-9-\./]+`

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).
**InternalServiceException**

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

**InvalidRequestException**

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

**NotFoundException**

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

**UnsupportedRegionException**

The requested operation is not supported in the region specified.

HTTP Status Code: 400

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
DeleteScalingPolicy

Deletes a fleet scaling policy. This action means that the policy is no longer in force and removes all record of it. To delete a scaling policy, specify both the scaling policy name and the fleet ID it is associated with.

To temporarily suspend scaling policies, call StopFleetActions (p. 219). This operation suspends all policies for the fleet.

Operations related to fleet capacity scaling include:

- DescribeFleetCapacity (p. 97)
- UpdateFleetCapacity (p. 238)
- DescribeEC2InstanceLimits (p. 90)
- Manage scaling policies:
  - PutScalingPolicy (p. 178) (auto-scaling)
  - DescribeScalingPolicies (p. 147) (auto-scaling)
  - DeleteScalingPolicy (p. 77) (auto-scaling)
- Manage fleet actions:
  - StartFleetActions (p. 201)
  - StopFleetActions (p. 219)

Request Syntax

```json
{
  "FleetId": "string",
  "Name": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.

**FleetId (p. 77)**

- Unique identifier for a fleet to be deleted.
- Type: String
- Pattern: ^fleet-\S+$
- Required: Yes

**Name (p. 77)**

- Descriptive label that is associated with a scaling policy. Policy names do not need to be unique.
- Type: String
Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

InternalServiceException

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

InvalidRequestException

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

NotFoundException

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

UnauthorizedException

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

Example

Delete a scaling policy

To delete a policy, we must specify both the fleet ID and name. The combination of these two values is what uniquely identifies the policy. Once a delete request is received, the policy is put into status DELETING and no longer affects the fleet's scaling activity.

HTTP requests are authenticated using an AWS Signature Version 4 signature in the Authorization header field.

Sample Request

```plaintext
POST / HTTP/1.1
Host: gamelift.us-west-2.amazonaws.com;
Accept-Encoding: identity
Content-Length: 336
User-Agent: aws-cli/1.11.36 Python/2.7.9 Windows/7 botocore/1.4.93
Content-Type: application/x-amz-json-1.0
```
Authorization: AWS4-HMAC-SHA256 Credential=A3IAIOSFODNN7EXAMPLE/20170406/us-west-2/gamelift/aws4_request, SignedHeaders=content-type;host;x-amz-date;x-amz-target, Signature=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
X-Amz-Date: 20170406T004805Z
X-Amz-Target: GameLift.DeleteScalingPolicy

{
   "FleetId": "fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
   "Name": "My_Target_Policy_1"
}

CLI syntax:

```bash
$aws gamelift delete-scaling-policy
--fleet-id "fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa"
--name "My_Target_Policy_1"
```

Sample Response

HTTP/1.1 200 OK undefined

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
DeleteVpcPeeringAuthorization

Cancels a pending VPC peering authorization for the specified VPC. If the authorization has already been used to create a peering connection, call DeleteVpcPeeringConnection (p. 82) to remove the connection.

VPC peering connection operations include:

- CreateVpcPeeringAuthorization (p. 56)
- DescribeVpcPeeringAuthorizations (p. 152)
- DeleteVpcPeeringAuthorization (p. 80)
- CreateVpcPeeringConnection (p. 61)
- DescribeVpcPeeringConnections (p. 154)
- DeleteVpcPeeringConnection (p. 82)

Request Syntax

```json
{
   "GameLiftAwsAccountId": "string",
   "PeerVpcId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.

**GameLiftAwsAccountId** (p. 80)

Unique identifier for the AWS account that you use to manage your Amazon GameLift fleet. You can find your Account ID in the AWS Management Console under account settings.

Type: String


Required: Yes

**PeerVpcId** (p. 80)

Unique identifier for a VPC with resources to be accessed by your Amazon GameLift fleet. The VPC must be in the same region where your fleet is deployed. To get VPC information, including IDs, use the Virtual Private Cloud service tools, including the VPC Dashboard in the AWS Management Console.

Type: String


Required: Yes
Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

InternalServiceException

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

InvalidRequestException

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

NotFoundException

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

UnauthorizedException

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
DeleteVpcPeeringConnection

Removes a VPC peering connection. To delete the connection, you must have a valid authorization for the VPC peering connection that you want to delete. You can check for an authorization by calling DescribeVpcPeeringAuthorizations (p. 152) or request a new one using CreateVpcPeeringAuthorization (p. 56).

Once a valid authorization exists, call this operation from the AWS account that is used to manage the Amazon GameLift fleets. Identify the connection to delete by the connection ID and fleet ID. If successful, the connection is removed.

VPC peering connection operations include:

- CreateVpcPeeringAuthorization (p. 56)
- DescribeVpcPeeringAuthorizations (p. 152)
- DeleteVpcPeeringAuthorization (p. 80)
- CreateVpcPeeringConnection (p. 61)
- DescribeVpcPeeringConnections (p. 154)
- DeleteVpcPeeringConnection (p. 82)

Request Syntax

```json
{
   "FleetId": "string",
   "VpcPeeringConnectionId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.

**FleetId (p. 82)**

Unique identifier for a fleet. This value must match the fleet ID referenced in the VPC peering connection record.

Type: String

Pattern: ^fleet-\S+

Required: Yes

**VpcPeeringConnectionId (p. 82)**

Unique identifier for a VPC peering connection. This value is included in the VpcPeeringConnection (p. 342) object, which can be retrieved by calling DescribeVpcPeeringConnections (p. 154).

Type: String
Required: Yes

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

**Errors**

For information about the errors that are common to all actions, see Common Errors (p. 347).

**InternalServiceException**

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

**InvalidRequestException**

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

**NotFoundException**

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

**UnauthorizedException**

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2

API Version 2015-10-01
DescribeAlias

Retrieves properties for an alias. This operation returns all alias metadata and settings. To get an alias's target fleet ID only, use ResolveAlias.

To get alias properties, specify the alias ID. If successful, the requested alias record is returned.

Alias-related operations include:

- CreateAlias (p. 10)
- ListAliases (p. 165)
- DescribeAlias (p. 84)
- UpdateAlias (p. 227)
- DeleteAlias (p. 65)
- ResolveAlias (p. 189)

Request Syntax

```
{
   "AliasId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**
In the following list, the required parameters are described first.

**AliasId (p. 84)**

Unique identifier for a fleet alias. Specify the alias you want to retrieve.

Type: String

Pattern: ^alias-\S+

Required: Yes

Response Syntax

```
{
   "Alias": {
      "AliasArn": "string",
      "AliasId": "string",
      "CreationTime": number,
      "Description": "string",
      "LastUpdatedTime": number,
      "Name": "string",
      "RoutingStrategy": {
```

API Version 2015-10-01
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**Alias (p. 84)**

Object that contains the requested alias.

Type: Alias (p. 266) object

**Errors**

For information about the errors that are common to all actions, see Common Errors (p. 347).

**InternalServiceException**

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

**InvalidRequestException**

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

**NotFoundException**

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

**UnauthorizedException**

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
See Also

- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
DescribeBuild

Retrieves properties for a build. To request a build record, specify a build ID. If successful, an object containing the build properties is returned.

Build-related operations include:

- CreateBuild (p. 13)
- ListBuilds (p. 168)
- DescribeBuild (p. 87)
- UpdateBuild (p. 230)
- DeleteBuild (p. 67)

Request Syntax

```
{
    "BuildId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.

**BuildId (p. 87)**

Unique identifier for a build to retrieve properties for.

Type: String

Pattern: ^build-\S+

Required: Yes

Response Syntax

```
{
    "Build": {
        "BuildId": "string",
        "CreationTime": number,
        "Name": "string",
        "OperatingSystem": "string",
        "SizeOnDisk": number,
        "Status": "string",
        "Version": "string"
    }
}
```
Response Elements

If the action is successful, the service sends back an HTTP 200 response. The following data is returned in JSON format by the service.

**Build** *(p. 87)*

Set of properties describing the requested build.

Type: **Build** *(p. 271)* object

Errors

For information about the errors that are common to all actions, see **Common Errors** *(p. 347).*

**InternalServiceException**

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

**InvalidRequestException**

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

**NotFoundException**

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

**UnauthorizedException**

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

Example

View a build record

This example retrieves the information, including current status, about a specific build.

HTTP requests are authenticated using an **AWS Signature Version 4** signature in the Authorization header field.

Sample Request

```
POST / HTTP/1.1
Host: gamelift.us-west-2.amazonaws.com;
Accept-Encoding: identity
Content-Length: 63
```
User-Agent: aws-cli/1.11.36 Python/2.7.9 Windows/7 botocore/1.4.93
Content-Type: application/x-amz-json-1.0
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20170406/us-west-2/gamelift/aws4_request, SignedHeaders=content-type;host;x-amz-date;x-amz-target, Signature=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
X-Amz-Date: 20170406T004805Z
X-Amz-Target: GameLift.DescribeBuild

{
  "BuildId": "build-1111aaaa-22bb-33cc-44dd-5555eeee66ff"
}

Sample Response

HTTP/1.1 200 OK
x-amzn-RequestId: b34f8665-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 310
Date: Thu, 06 Apr 2017 00:48:07 GMT

{
  "Build": {
    "BuildId": "build-1111aaaa-22bb-33cc-44dd-5555eeee66ff",
    "CreationTime": 1496708916.18,
    "Name": "My_Game_Server_Build_One",
    "OperatingSystem": "AMAZON_LINUX",
    "SizeOnDisk": 1304924,
    "Status": "READY",
    "Version": "12345.678"
  }
}

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
DescribeEC2InstanceLimits

Retrieves the following information for the specified EC2 instance type:

- maximum number of instances allowed per AWS account (service limit)
- current usage level for the AWS account

Service limits vary depending on region. Available regions for Amazon GameLift can be found in the AWS Management Console for Amazon GameLift (see the drop-down list in the upper right corner).

Fleet-related operations include:

- CreateFleet (p. 19)
- ListFleets (p. 173)
- DeleteFleet (p. 70)
- Describe fleets:
  - DescribeFleetAttributes (p. 93)
  - DescribeFleetCapacity (p. 97)
  - DescribeFleetPortSettings (p. 107)
  - DescribeFleetUtilization (p. 110)
  - DescribeRuntimeConfiguration (p. 144)
  - DescribeEC2InstanceLimits (p. 90)
  - DescribeFleetEvents (p. 101)
- Update fleets:
  - UpdateFleetAttributes (p. 234)
  - UpdateFleetCapacity (p. 238)
  - UpdateFleetPortSettings (p. 242)
  - UpdateRuntimeConfiguration (p. 258)
- Manage fleet actions:
  - StartFleetActions (p. 201)
  - StopFleetActions (p. 219)

Request Syntax

```json
{
   "EC2InstanceType": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.
EC2InstanceType (p. 90)

Name of an EC2 instance type that is supported in Amazon GameLift. A fleet instance type determines the computing resources of each instance in the fleet, including CPU, memory, storage, and networking capacity. Amazon GameLift supports the following EC2 instance types. See Amazon EC2 Instance Types for detailed descriptions. Leave this parameter blank to retrieve limits for all types.

Type: String

Valid Values:
- t2.micro
- t2.small
- t2.medium
- t2.large
- c3.large
- c3.xlarge
- c3.2xlarge
- c3.4xlarge
- c3.8xlarge
- c4.large
- c4.xlarge
- c4.2xlarge
- c4.4xlarge
- c4.8xlarge
- r3.large
- r3.xlarge
- r3.2xlarge
- r3.8xlarge
- r4.large
- r4.xlarge
- r4.2xlarge
- r4.4xlarge
- r4.8xlarge
- r4.16xlarge
- m3.medium
- m3.large
- m3.xlarge
- m3.2xlarge
- m4.large
- m4.xlarge
- m4.2xlarge
- m4.4xlarge
- m4.10xlarge

Required: No

Response Syntax

```json
{
    "EC2InstanceLimits": [
        {
            "CurrentInstances": number,
            "EC2InstanceType": "string",
            "InstanceLimit": number
        }
    ]
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

EC2InstanceLimits (p. 91)

Object that contains the maximum number of instances for the specified instance type.

Type: Array of EC2InstanceLimit (p. 276) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

InternalServiceException

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

InvalidRequestException

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.
HTTP Status Code: 400

**UnauthorizedException**

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
DescribeFleetAttributes

Retrieves fleet properties, including metadata, status, and configuration, for one or more fleets. You can request attributes for all fleets, or specify a list of one or more fleet IDs. When requesting multiple fleets, use the pagination parameters to retrieve results as a set of sequential pages. If successful, a FleetAttributes (p. 280) object is returned for each requested fleet ID. When specifying a list of fleet IDs, attribute objects are returned only for fleets that currently exist.

Note
Some API actions may limit the number of fleet IDs allowed in one request. If a request exceeds this limit, the request fails and the error message includes the maximum allowed.

Fleet-related operations include:

- CreateFleet (p. 19)
- ListFleets (p. 173)
- DeleteFleet (p. 70)
- Describe fleets:
  - DescribeFleetAttributes (p. 93)
  - DescribeFleetCapacity (p. 97)
  - DescribeFleetPortSettings (p. 107)
  - DescribeFleetUtilization (p. 110)
  - DescribeRuntimeConfiguration (p. 144)
  - DescribeEC2InstanceLimits (p. 90)
  - DescribeFleetEvents (p. 101)
- Update fleets:
  - UpdateFleetAttributes (p. 234)
  - UpdateFleetCapacity (p. 238)
  - UpdateFleetPortSettings (p. 242)
  - UpdateRuntimeConfiguration (p. 258)
- Manage fleet actions:
  - StartFleetActions (p. 201)
  - StopFleetActions (p. 219)

Request Syntax

```json
{
  "FleetIds": [ "string" ],
  "Limit": number,
  "NextToken": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

Note
In the following list, the required parameters are described first.
FleetIds (p. 93)

Unique identifier for a fleet(s) to retrieve attributes for. To request attributes for all fleets, leave this parameter empty.

Type: Array of strings

Array Members: Minimum number of 1 item.

Pattern: ^fleet-\S+

Required: No

Limit (p. 93)

Maximum number of results to return. Use this parameter with NextToken to get results as a set of sequential pages. This parameter is ignored when the request specifies one or a list of fleet IDs.

Type: Integer

Valid Range: Minimum value of 1.

Required: No

NextToken (p. 93)

Token that indicates the start of the next sequential page of results. Use the token that is returned with a previous call to this action. To start at the beginning of the result set, do not specify a value. This parameter is ignored when the request specifies one or a list of fleet IDs.

Type: String


Required: No

Response Syntax

```
{
  "FleetAttributes": [
    {
      "BuildId": "string",
      "CreationTime": number,
      "Description": "string",
      "FleetArn": "string",
      "FleetId": "string",
      "FleetType": "string",
      "InstanceType": "string",
      "LogPaths": [ "string" ],
      "MetricGroups": [ "string" ],
      "Name": "string",
      "NewGameSessionProtectionPolicy": "string",
      "OperatingSystem": "string",
      "ResourceCreationLimitPolicy": {
        "NewGameSessionsPerCreator": number,
        "PolicyPeriodInMinutes": number
      },
      "ServerLaunchParameters": "string",
      "ServerLaunchPath": "string",
      "Status": "string",
      "StoppedActions": [ "string" ],
      "TerminationTime": number
    }
  ]
}
```
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

FleetAttributes (p. 94)
Collection of objects containing attribute metadata for each requested fleet ID.
Type: Array of FleetAttributes (p. 280) objects

NextToken (p. 94)
Token that indicates where to resume retrieving results on the next call to this action. If no token is returned, these results represent the end of the list.
Type: String

Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

InternalServiceException
The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.
HTTP Status Code: 500

InvalidRequestException
One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.
HTTP Status Code: 400

NotFoundException
A service resource associated with the request could not be found. Clients should not retry such requests.
HTTP Status Code: 400

UnauthorizedException
The client failed authentication. Clients should not retry such requests.
HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:
See Also

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
DescribeFleetCapacity

Retrieves the current status of fleet capacity for one or more fleets. This information includes the number of instances that have been requested for the fleet and the number currently active. You can request capacity for all fleets, or specify a list of one or more fleet IDs. When requesting multiple fleets, use the pagination parameters to retrieve results as a set of sequential pages. If successful, a FleetCapacity (p. 285) object is returned for each requested fleet ID. When specifying a list of fleet IDs, attribute objects are returned only for fleets that currently exist.

**Note**
Some API actions may limit the number of fleet IDs allowed in one request. If a request exceeds this limit, the request fails and the error message includes the maximum allowed.

Fleet-related operations include:

- CreateFleet (p. 19)
- ListFleets (p. 173)
- DeleteFleet (p. 70)

**Describe fleets:**
- DescribeFleetAttributes (p. 93)
- DescribeFleetCapacity (p. 97)
- DescribeFleetPortSettings (p. 107)
- DescribeFleetUtilization (p. 110)
- DescribeRuntimeConfiguration (p. 144)
- DescribeEC2InstanceLimits (p. 90)
- DescribeFleetEvents (p. 101)

**Update fleets:**
- UpdateFleetAttributes (p. 234)
- UpdateFleetCapacity (p. 238)
- UpdateFleetPortSettings (p. 242)
- UpdateRuntimeConfiguration (p. 258)

**Manage fleet actions:**
- StartFleetActions (p. 201)
- StopFleetActions (p. 219)

**Request Syntax**

```json
{
  "FleetIds": [ "string" ],
  "Limit": number,
  "NextToken": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.
Note
In the following list, the required parameters are described first.

**FleetIds (p. 97)**
Unique identifier for a fleet(s) to retrieve capacity information for. To request capacity information for all fleets, leave this parameter empty.

Type: Array of strings

Array Members: Minimum number of 1 item.

Pattern: ^fleet-\S+

Required: No

**Limit (p. 97)**
Maximum number of results to return. Use this parameter with **NextToken** to get results as a set of sequential pages. This parameter is ignored when the request specifies one or a list of fleet IDs.

Type: Integer

Valid Range: Minimum value of 1.

Required: No

**NextToken (p. 97)**
Token that indicates the start of the next sequential page of results. Use the token that is returned with a previous call to this action. To start at the beginning of the result set, do not specify a value. This parameter is ignored when the request specifies one or a list of fleet IDs.

Type: String

Required: No

**Response Syntax**

```json
{
    "FleetCapacity": [
        {
            "FleetId": "string",
            "InstanceCounts": {
                "ACTIVE": number,
                "DESIRED": number,
                "IDLE": number,
                "MAXIMUM": number,
                "MINIMUM": number,
                "PENDING": number,
                "TERMINATING": number
            },
            "InstanceType": "string"
        }
        ,
        "NextToken": "string"
    }
}
```
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**FleetCapacity (p. 98)**

Collection of objects containing capacity information for each requested fleet ID. Leave this parameter empty to retrieve capacity information for all fleets.

Type: Array of FleetCapacity (p. 285) objects

**NextToken (p. 98)**

Token that indicates where to resume retrieving results on the next call to this action. If no token is returned, these results represent the end of the list.

Type: String


Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

**InternalServiceException**

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

**InvalidRequestException**

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

**NotFoundException**

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

**UnauthorizedException**

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V2
DescribeFleetEvents

Retrieves entries from the specified fleet's event log. You can specify a time range to limit the result set. Use the pagination parameters to retrieve results as a set of sequential pages. If successful, a collection of event log entries matching the request are returned.

Fleet-related operations include:

- CreateFleet (p. 19)
- ListFleets (p. 173)
- DeleteFleet (p. 70)
- Describe fleets:
  - DescribeFleetAttributes (p. 93)
  - DescribeFleetCapacity (p. 97)
  - DescribeFleetPortSettings (p. 107)
  - DescribeFleetUtilization (p. 110)
  - DescribeRuntimeConfiguration (p. 144)
  - DescribeEC2InstanceLimits (p. 90)
  - DescribeFleetEvents (p. 101)
- Update fleets:
  - UpdateFleetAttributes (p. 234)
  - UpdateFleetCapacity (p. 238)
  - UpdateFleetPortSettings (p. 242)
  - UpdateRuntimeConfiguration (p. 258)
- Manage fleet actions:
  - StartFleetActions (p. 201)
  - StopFleetActions (p. 219)

Request Syntax

```
{
  "EndTime": number,
  "FleetId": "string",
  "Limit": number,
  "NextToken": "string",
  "StartTime": number
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.

**FleetId (p. 101)**

Unique identifier for a fleet to get event logs for.
Type: String

Pattern: ^fleet-\S+

Required: Yes

EndTime (p. 101)

Most recent date to retrieve event logs for. If no end time is specified, this call returns entries from
the specified start time up to the present. Format is a number expressed in Unix time as milliseconds
(ex: "1469498468.057").

Type: Timestamp

Required: No

Limit (p. 101)

Maximum number of results to return. Use this parameter with NextToken to get results as a set of
sequential pages.

Type: Integer

Valid Range: Minimum value of 1.

Required: No

NextToken (p. 101)

Token that indicates the start of the next sequential page of results. Use the token that is returned
with a previous call to this action. To start at the beginning of the result set, do not specify a value.

Type: String


Required: No

StartTime (p. 101)

Earliest date to retrieve event logs for. If no start time is specified, this call returns entries starting
from when the fleet was created to the specified end time. Format is a number expressed in Unix
time as milliseconds (ex: "1469498468.057").

Type: Timestamp

Required: No

Response Syntax

```json
{
    "Events": [
    {
        "EventCode": "string",
        "EventId": "string",
        "EventTime": number,
        "Message": "string",
        "PreSignedLogUrl": "string",
        "ResourceId": "string"
    }
    ],
    "NextToken": "string"
}
```
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

Events (p. 102)

Collection of objects containing event log entries for the specified fleet.

Type: Array of Event (p. 277) objects

NextToken (p. 102)

Token that indicates where to resume retrieving results on the next call to this action. If no token is returned, these results represent the end of the list.

Type: String


Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

InternalServiceException

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

InvalidRequestException

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

NotFoundException

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

UnauthorizedException

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

Example

Request events for a limited time span

This example returns all events that occurred for the fleet during the specified time span. In this example, the time span is on June 21, 2017, from 15:45:00 to 16:15:00 (local time).
HTTP requests are authenticated using an **AWS Signature Version 4** signature in the Authorization header field.

**Sample Request**

```plaintext
POST / HTTP/1.1
Host: gamelift.us-west-2.amazonaws.com;
Accept-Encoding: identity
Content-Length: 109
User-Agent: aws-cli/1.11.36 Python/2.7.9 Windows/7 botocore/1.4.93
Authorization: AWS4-HMAC-SHA256  Credential=AKIAIOSFODNN7EXAMPLE/20170406/us-west-2/ 
gamelift/aws4_request, SignedHeaders=content-type;host;x-amz-date;x-amz-target, 
Signature=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
X-Amz-Date: 20170406T004805Z
X-Amz-Target: GameLift.DescribeFleetEvents

{"FleetId": "fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa", 
"StartTime": 1498085100,
"EndTime": 1498086900}
```

**Sample Response**

```plaintext
HTTP/1.1 200 OK
x-amzn-RequestId: b34f8665-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 4481
Date: Thu, 06 Apr 2017 00:48:07 GMT

{"Events": [
    {
      "EventCode": "FLEET_SCALING_EVENT",
      "EventId": "494e4384-49a3-4185-8cd0-1b78417af576",
      "EventTime": 1498086288.747,
      "Message": "Fleet fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa capacity settings have been changed: Desired size = 1, Max size = 1, Min size = 1",
      "ResourceId": "fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa"},
    {
      "EventCode": "FLEET_STATE_ACTIVE",
      "EventId": "3b7474c7-f687-4491-84ad-6be6554a3b1b",
      "EventTime": 1498086210.461,
      "Message": "Fleet fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa changed state to ACTIVE",
      "ResourceId": "fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa"},
    {
      "EventCode": "FLEET_STATE_ACTIVATING",
      "EventId": "94b66df4-3d3d-4e5b-bbf7-3c152bb21270",
      "EventTime": 1498085769.755,
      "Message": "Fleet fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa changed state to ACTIVATING",
      "ResourceId": "fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa"},
    {
      "EventCode": "FLEET_STATE_BUILDING",
      "EventId": "6b085682-3d3d-4e5b-bbf7-3c152bb21270",
      "EventTime": 1498085757.653,
      "Message": "Fleet fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa changed state to BUILDING",
      "ResourceId": "fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa"},
    {
      "EventCode": "FLEET_STATE_VALIDATING",
      "EventId": "9cfb253a-bee8-415a-bee0-d656779e139",
      "EventTime": 1498085690.528,
      "Message": "Fleet fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa changed state to VALIDATING",
      "ResourceId": "fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa"},
    {
      "EventCode": "FLEET_CREATION_VALIDATING_RUNTIME_CONFIG",
      "EventId": "b1b74f8-b0e8-45df-a075-f4e3a2d1248c",
      "EventTime": 1498085354.858,
      "Message": "Fleet fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa changed state to 
      "ResourceId": "fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa"
    }
]`
```
See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
DescribeFleetPortSettings

Retrieves the inbound connection permissions for a fleet. Connection permissions include a range of IP addresses and port settings that incoming traffic can use to access server processes in the fleet. To get a fleet's inbound connection permissions, specify a fleet ID. If successful, a collection of IpPermission (p. 309) objects is returned for the requested fleet ID. If the requested fleet has been deleted, the result set is empty.

Fleet-related operations include:

- CreateFleet (p. 19)
- ListFleets (p. 173)
- DeleteFleet (p. 70)
- Describe fleets:
  - DescribeFleetAttributes (p. 93)
  - DescribeFleetCapacity (p. 97)
  - DescribeFleetPortSettings (p. 107)
  - DescribeFleetUtilization (p. 110)
  - DescribeRuntimeConfiguration (p. 144)
  - DescribeEC2InstanceLimits (p. 90)
  - DescribeFleetEvents (p. 101)
- Update fleets:
  - UpdateFleetAttributes (p. 234)
  - UpdateFleetCapacity (p. 238)
  - UpdateFleetPortSettings (p. 242)
  - UpdateRuntimeConfiguration (p. 258)
- Manage fleet actions:
  - StartFleetActions (p. 201)
  - StopFleetActions (p. 219)

Request Syntax

```json
{
    "FleetId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.

**FleetId (p. 107)**

Unique identifier for a fleet to retrieve port settings for.

Type: String
Response Syntax

```
{
    "InboundPermissions": [
        {
            "FromPort": number,
            "IpRange": "string",
            "Protocol": "string",
            "ToPort": number
        }
    ]
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**InboundPermissions (p. 108)**

Object that contains port settings for the requested fleet ID.

Type: Array of IpPermission (p. 309) objects

Array Members: Maximum number of 50 items.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

**InternalServiceException**

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

**InvalidRequestException**

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

**NotFoundException**

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

**UnauthorizedException**

The client failed authentication. Clients should not retry such requests.
HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
DescribeFleetUtilization

Retrieves utilization statistics for one or more fleets. You can request utilization data for all fleets, or specify a list of one or more fleet IDs. When requesting multiple fleets, use the pagination parameters to retrieve results as a set of sequential pages. If successful, a FleetUtilization (p. 287) object is returned for each requested fleet ID. When specifying a list of fleet IDs, utilization objects are returned only for fleets that currently exist.

**Note**

Some API actions may limit the number of fleet IDs allowed in one request. If a request exceeds this limit, the request fails and the error message includes the maximum allowed.

Fleet-related operations include:

- CreateFleet (p. 19)
- ListFleets (p. 173)
- DeleteFleet (p. 70)
- Describe fleets:
  - DescribeFleetAttributes (p. 93)
  - DescribeFleetCapacity (p. 97)
  - DescribeFleetPortSettings (p. 107)
  - DescribeFleetUtilization (p. 110)
  - DescribeRuntimeConfiguration (p. 144)
  - DescribeEC2InstanceLimits (p. 90)
  - DescribeFleetEvents (p. 101)
- Update fleets:
  - UpdateFleetAttributes (p. 234)
  - UpdateFleetCapacity (p. 238)
  - UpdateFleetPortSettings (p. 242)
  - UpdateRuntimeConfiguration (p. 258)
- Manage fleet actions:
  - StartFleetActions (p. 201)
  - StopFleetActions (p. 219)

**Request Syntax**

```json
{
   "FleetIds": [ "string" ],
   "Limit": number,
   "NextToken": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see [Common Parameters (p. 345)].

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.
FleetIds (p. 110)

Unique identifier for a fleet(s) to retrieve utilization data for. To request utilization data for all fleets, leave this parameter empty.

Type: Array of strings

Array Members: Minimum number of 1 item.

Pattern: ^fleet-\S+

Required: No

Limit (p. 110)

Maximum number of results to return. Use this parameter with NextToken to get results as a set of sequential pages. This parameter is ignored when the request specifies one or a list of fleet IDs.

Type: Integer

Valid Range: Minimum value of 1.

Required: No

NextToken (p. 110)

Token that indicates the start of the next sequential page of results. Use the token that is returned with a previous call to this action. To start at the beginning of the result set, do not specify a value. This parameter is ignored when the request specifies one or a list of fleet IDs.

Type: String


Required: No

Response Syntax

```json
{
   "FleetUtilization": [ 
      {
         "ActiveGameSessionCount": number,
         "ActiveServerProcessCount": number,
         "CurrentPlayerSessionCount": number,
         "FleetId": "string",
         "MaximumPlayerSessionCount": number
      }
   ],
   "NextToken": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

FleetUtilization (p. 111)

Collection of objects containing utilization information for each requested fleet ID.
Type: Array of FleetUtilization (p. 287) objects

NextToken (p. 111)

Token that indicates where to resume retrieving results on the next call to this action. If no token is returned, these results represent the end of the list.

Type: String


Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

InternalServiceException

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

InvalidRequestException

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

NotFoundException

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

UnauthorizedException

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
DescribeGameSessionDetails

Retrieves properties, including the protection policy in force, for one or more game sessions. This action can be used in several ways: (1) provide a GameSessionId or GameSessionArn to request details for a specific game session; (2) provide either a FleetId or an AliasId to request properties for all game sessions running on a fleet.

To get game session record(s), specify just one of the following: game session ID, fleet ID, or alias ID. You can filter this request by game session status. Use the pagination parameters to retrieve results as a set of sequential pages. If successful, a GameSessionDetail (p. 296) object is returned for each session matching the request.

Game-session-related operations include:

- CreateGameSession (p. 31)
- DescribeGameSessions (p. 123)
- DescribeGameSessionDetails (p. 113)
- SearchGameSessions (p. 192)
- UpdateGameSession (p. 246)
- GetGameSessionLogUrl (p. 157)
- Game session placements
  - StartGameSessionPlacement (p. 204)
  - DescribeGameSessionPlacement (p. 117)
  - StopGameSessionPlacement (p. 222)

Request Syntax

```json
{
  "AliasId": "string",
  "FleetId": "string",
  "GameSessionId": "string",
  "Limit": number,
  "NextToken": "string",
  "StatusFilter": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.

**AliasId (p. 113)**

Unique identifier for an alias associated with the fleet to retrieve all game sessions for.

Type: String

Pattern: ^alias-\S+$
Required: No

**FleetId (p. 113)**

Unique identifier for a fleet to retrieve all game sessions active on the fleet.

Type: String

Pattern: ^fleet-\S+$

Required: No

**GameSessionId (p. 113)**

Unique identifier for the game session to retrieve.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: [a-zA-Z0-9:/-]+$

Required: No

**Limit (p. 113)**

Maximum number of results to return. Use this parameter with NextToken to get results as a set of sequential pages.

Type: Integer

Valid Range: Minimum value of 1.

Required: No

**NextToken (p. 113)**

Token that indicates the start of the next sequential page of results. Use the token that is returned with a previous call to this action. To start at the beginning of the result set, do not specify a value.

Type: String


Required: No

**StatusFilter (p. 113)**

Game session status to filter results on. Possible game session statuses include ACTIVE, TERMINATED, ACTIVATING and TERMINATING (the last two are transitory).

Type: String


Required: No

---

**Response Syntax**

```json
{
    "GameSessionDetails": [
    
    ]
}  
```

API Version 2015-10-01
114
"GameSession": {  
    "CreationTime": number,  
    "CreatorId": "string",  
    "CurrentPlayerSessionCount": number,  
    "FleetId": "string",  
    "GameProperties": [  
        {  
            "Key": "string",  
            "Value": "string"  
        }  
    ],  
    "GameSessionData": "string",  
    "GameSessionId": "string",  
    "IpAddress": "string",  
    "MatchmakerData": "string",  
    "MaximumPlayerSessionCount": number,  
    "Name": "string",  
    "PlayerSessionCreationPolicy": "string",  
    "Port": number,  
    "Status": "string",  
    "StatusReason": "string",  
    "TerminationTime": number  
},  
"ProtectionPolicy": "string"  
},  
"NextToken": "string"  
}  

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

GameSessionDetails (p. 114)

Collection of objects containing game session properties and the protection policy currently in force for each session matching the request.

Type: Array of GameSessionDetail (p. 296) objects

NextToken (p. 114)

Token that indicates where to resume retrieving results on the next call to this action. If no token is returned, these results represent the end of the list.

Type: String


Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

InternalServiceException

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500
InvalidRequestException

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

NotFoundException

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

TerminalRoutingStrategyException

The service is unable to resolve the routing for a particular alias because it has a terminal RoutingStrategy (p. 329) associated with it. The message returned in this exception is the message defined in the routing strategy itself. Such requests should only be retried if the routing strategy for the specified alias is modified.

HTTP Status Code: 400

UnauthorizedException

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
DescribeGameSessionPlacement

Retrieves properties and current status of a game session placement request. To get game session placement details, specify the placement ID. If successful, a GameSessionPlacement (p. 297) object is returned.

Game-session-related operations include:

- CreateGameSession (p. 31)
- DescribeGameSessions (p. 123)
- DescribeGameSessionDetails (p. 113)
- SearchGameSessions (p. 192)
- UpdateGameSession (p. 246)
- GetGameSessionLogUrl (p. 157)
- Game session placements
  - StartGameSessionPlacement (p. 204)
  - DescribeGameSessionPlacement (p. 117)
  - StopGameSessionPlacement (p. 222)

Request Syntax

```json
{
    "PlacementId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.

**PlacementId (p. 117)**

Unique identifier for a game session placement to retrieve.

Type: String


Pattern: `[a-zA-Z0-9-]+`

Required: Yes

Response Syntax

```json
{
    "GameSessionPlacement": {
        "EndTime": number,
    }
}
```
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

GameSessionPlacement (p. 117)

Object that describes the requested game session placement.

Type: GameSessionPlacement (p. 297) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

InternalServiceException

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

InvalidRequestException

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.
HTTP Status Code: 400

`NotFoundException`

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

`UnauthorizedException`

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
DescribeGameSessionQueues

Retrieves the properties for one or more game session queues. When requesting multiple queues, use the pagination parameters to retrieve results as a set of sequential pages. If successful, a GameSessionQueue (p. 301) object is returned for each requested queue. When specifying a list of queues, objects are returned only for queues that currently exist in the region.

Queue-related operations include:

- CreateGameSessionQueue (p. 37)
- DescribeGameSessionQueues (p. 120)
- UpdateGameSessionQueue (p. 250)
- DeleteGameSessionQueue (p. 73)

Request Syntax

```json
{
    "Limit": number,
    "Names": [ "string" ],
    "NextToken": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.

**Limit (p. 120)**

Maximum number of results to return. Use this parameter with NextToken to get results as a set of sequential pages.

Type: Integer

Valid Range: Minimum value of 1.

Required: No

**Names (p. 120)**

List of queue names to retrieve information for. To request settings for all queues, leave this parameter empty.

Type: Array of strings


Pattern: [ a-zA-Z0-9- ]+

Required: No
NextToken (p. 120)

Token that indicates the start of the next sequential page of results. Use the token that is returned with a previous call to this action. To start at the beginning of the result set, do not specify a value.

Type: String


Required: No

Response Syntax

```
{
   "GameSessionQueues": [
      {
         "Destinations": [
            {
               "DestinationArn": "string"
            }
         ],
         "GameSessionQueueArn": "string",
         "Name": "string",
         "PlayerLatencyPolicies": [
            {
               "MaximumIndividualPlayerLatencyMilliseconds": number,
               "PolicyDurationSeconds": number
            }
         ],
         "TimeoutInSeconds": number
      }
   ],
   "NextToken": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

GameSessionQueues (p. 121)

Collection of objects that describes the requested game session queues.

Type: Array of GameSessionQueue (p. 301) objects

NextToken (p. 121)

Token that indicates where to resume retrieving results on the next call to this action. If no token is returned, these results represent the end of the list.

Type: String


Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).
InternalServiceException

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

InvalidRequestException

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

NotFoundException

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

UnauthorizedException

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
DescribeGameSessions

Retrieves a set of one or more game sessions. Request a specific game session or request all game sessions on a fleet. Alternatively, use SearchGameSessions (p. 192) to request a set of active game sessions that are filtered by certain criteria. To retrieve protection policy settings for game sessions, use DescribeGameSessionDetails (p. 113).

To get game sessions, specify one of the following: game session ID, fleet ID, or alias ID. You can filter this request by game session status. Use the pagination parameters to retrieve results as a set of sequential pages. If successful, a GameSession (p. 290) object is returned for each game session matching the request.

Available in Amazon GameLift Local.

Game-session-related operations include:

- CreateGameSession (p. 31)
- DescribeGameSessions (p. 123)
- DescribeGameSessionDetails (p. 113)
- SearchGameSessions (p. 192)
- UpdateGameSession (p. 246)
- GetGameSessionLogUrl (p. 157)
- Game session placements
  - StartGameSessionPlacement (p. 204)
  - DescribeGameSessionPlacement (p. 117)
  - StopGameSessionPlacement (p. 222)

Request Syntax

```json
{
  "AliasId": "string",
  "FleetId": "string",
  "GameSessionId": "string",
  "Limit": number,
  "NextToken": "string",
  "StatusFilter": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**
In the following list, the required parameters are described first.

**AliasId (p. 123)**

Unique identifier for an alias associated with the fleet to retrieve all game sessions for.

Type: String
**Response Syntax**

```json
{
    "GameSessions": [
        {
            // Fields for GameSessions go here
        }
    ]
}
```
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**GameSessions (p. 124)**

Collection of objects containing game session properties for each session matching the request.

Type: Array of GameSession (p. 290) objects

**NextToken (p. 124)**

Token that indicates where to resume retrieving results on the next call to this action. If no token is returned, these results represent the end of the list.

Type: String


Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

**InternalServiceException**

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

**InvalidRequestException**

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.
HTTP Status Code: 400

**NotFoundException**

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

**TerminalRoutingStrategyException**

The service is unable to resolve the routing for a particular alias because it has a terminal `RoutingStrategy` (p. 329) associated with it. The message returned in this exception is the message defined in the routing strategy itself. Such requests should only be retried if the routing strategy for the specified alias is modified.

HTTP Status Code: 400

**UnauthorizedException**

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
DescribeInstances

Retrieves information about a fleet's instances, including instance IDs. Use this action to get details on all instances in the fleet or get details on one specific instance.

To get a specific instance, specify fleet ID and instance ID. To get all instances in a fleet, specify a fleet ID only. Use the pagination parameters to retrieve results as a set of sequential pages. If successful, an Instance (p. 304) object is returned for each result.

Request Syntax

```
{
    "FleetId": "string",
    "InstanceId": "string",
    "Limit": number,
    "NextToken": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.

**FleetId (p. 127)**

Unique identifier for a fleet to retrieve instance information for.

Type: String

Pattern: ^fleet-\S+

Required: Yes

**InstanceId (p. 127)**

Unique identifier for an instance to retrieve. Specify an instance ID or leave blank to retrieve all instances in the fleet.

Type: String

Pattern: [a-zA-Z0-9.-]+

Required: No

**Limit (p. 127)**

Maximum number of results to return. Use this parameter with NextToken to get results as a set of sequential pages.

Type: Integer

Valid Range: Minimum value of 1.

Required: No
NextToken (p. 127)

Token that indicates the start of the next sequential page of results. Use the token that is returned with a previous call to this action. To start at the beginning of the result set, do not specify a value.

Type: String

Required: No

Response Syntax

```
{
  "Instances": [
    {
      "CreationTime": number,
      "FleetId": "string",
      "InstanceId": "string",
      "IpAddress": "string",
      "OperatingSystem": "string",
      "Status": "string",
      "Type": "string"
    }
  ],
  "NextToken": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

Instances (p. 128)

Collection of objects containing properties for each instance returned.

Type: Array of Instance (p. 304) objects

NextToken (p. 128)

Token that indicates where to resume retrieving results on the next call to this action. If no token is returned, these results represent the end of the list.

Type: String


Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

InternalServiceException

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.
HTTP Status Code: 500
InvalidRequestException

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400
NotFoundException

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400
UnauthorizedException

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

Example

Get data on a fleet instance

This example requests information on one instance in a fleet. Instances are not returned in any particular order. If you want to get data on all instances in a fleet, omit the Limit parameter.

HTTP requests are authenticated using an AWS Signature Version 4 signature in the Authorization header field.

Sample Request

```
POST / HTTP/1.1
Host: gamelift.us-west-2.amazonaws.com;
Accept-Encoding: identity
Content-Length: 105
User-Agent: aws-cli/1.11.36 Python/2.7.9 Windows/7 botocore/1.4.93
Content-Type: application/x-amz-json-1.0
Authorization: AWS4-HMAC-SHA256  Credential=AKIAIOSFODNN7EXAMPLE/20170406/us-west-2/
gamelift/aws4_request, SignedHeaders=content-type;host;x-amz-date;x-amz-target,
Signature=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
X-Amz-Date: 20170406T004805Z
X-Amz-Target: GameLift.DescribeInstances

{"FleetId": "fleet-9999ffff-88ee-77dd-66cc-5555bbbb44aa",
"Limit": "1"}
```

CLI syntax:
```
aws gamelift describe-instances --fleet-id "fleet-9999ffff-88ee-77dd-66cc-5555bbbb44aa"
--limit 1
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: b34f8665-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 1986
```
See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
DescribeMatchmaking

Retrieves one or more matchmaking tickets. Use this operation to retrieve ticket information, including status and—once a successful match is made—acquire connection information for the resulting new game session.

You can use this operation to track the progress of matchmaking requests (through polling) as an alternative to using event notifications. See more details on tracking matchmaking requests through polling or notifications in StartMatchmaking (p. 214).

To request matchmaking tickets, provide a list of up to 10 ticket IDs. If the request is successful, a ticket object is returned for each requested ID that currently exists.

Matchmaking-related operations include:

- StartMatchmaking (p. 214)
- DescribeMatchmaking (p. 131)
- StopMatchmaking (p. 225)
- AcceptMatch (p. 7)
- StartMatchBackfill (p. 210)

Request Syntax

```
{
  "TicketIds": [ "string" ]
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.

**TicketIds (p. 131)**

Unique identifier for a matchmaking ticket. You can include up to 10 ID values.

Type: Array of strings


Pattern: `[a-zA-Z0-9-\.]`

Required: Yes

Response Syntax

```
{
  "TicketList": [
    {
      "ConfigurationName": "string",
      
    }
  ]
}
```
Response Elements

If the action is successful, the service sends back an HTTP 200 response. The following data is returned in JSON format by the service.

**TicketList (p. 131)**

Collection of existing matchmaking ticket objects matching the request.

Type: Array of [MatchmakingTicket (p. 317)] objects

**Errors**

For information about the errors that are common to all actions, see [Common Errors (p. 347)].

**InternalServiceException**

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.
HTTP Status Code: 500
**InvalidRequestException**

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400
**UnsupportedRegionException**

The requested operation is not supported in the region specified.

HTTP Status Code: 400

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
DescribeMatchmakingConfigurations

Retrieves the details of FlexMatch matchmaking configurations. With this operation, you have the following options: (1) retrieve all existing configurations, (2) provide the names of one or more configurations to retrieve, or (3) retrieve all configurations that use a specified rule set name. When requesting multiple items, use the pagination parameters to retrieve results as a set of sequential pages. If successful, a configuration is returned for each requested name. When specifying a list of names, only configurations that currently exist are returned.

Operations related to match configurations and rule sets include:

- CreateMatchmakingConfiguration (p. 41)
- DescribeMatchmakingConfigurations (p. 134)
- UpdateMatchmakingConfiguration (p. 253)
- DeleteMatchmakingConfiguration (p. 75)
- CreateMatchmakingRuleSet (p. 46)
- DescribeMatchmakingRuleSets (p. 137)
- ValidateMatchmakingRuleSet (p. 262)

Request Syntax

```
{
    "Limit": number,
    "Names": [ "string" ],
    "NextToken": "string",
    "RuleSetName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.

**Limit (p. 134)**

Maximum number of results to return. Use this parameter with NextToken to get results as a set of sequential pages. This parameter is limited to 10.

Type: Integer

Valid Range: Minimum value of 1.

Required: No

**Names (p. 134)**

Unique identifier for a matchmaking configuration(s) to retrieve. To request all existing configurations, leave this parameter empty.

Type: Array of strings
Pattern: [a-zA-Z0-9-\.]+
Required: No

**NextToken (p. 134)**

Token that indicates the start of the next sequential page of results. Use the token that is returned with a previous call to this action. To start at the beginning of the result set, do not specify a value.

Type: String
Required: No

**RuleSetName (p. 134)**

Unique identifier for a matchmaking rule set. Use this parameter to retrieve all matchmaking configurations that use this rule set.

Type: String
Pattern: [a-zA-Z0-9-\.]+
Required: No

---

**Response Syntax**

```json
{
  "Configurations": [
    {
      "AcceptanceRequired": boolean,
      "AcceptanceTimeoutSeconds": number,
      "AdditionalPlayerCount": number,
      "CreationTime": number,
      "CustomEventData": "string",
      "Description": "string",
      "GameProperties": [
        {
          "Key": "string",
          "Value": "string"
        }
      ],
      "GameSessionData": "string",
      "GameSessionQueueArns": [ "string" ],
      "Name": "string",
      "NotificationTarget": "string",
      "RequestTimeoutSeconds": number,
      "RuleSetName": "string"
    }
  ],
  "NextToken": "string"
}
```

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response.
The following data is returned in JSON format by the service.

**Configurations (p. 135)**

Collection of requested matchmaking configuration objects.

Type: Array of MatchmakingConfiguration (p. 312) objects

**NextToken (p. 135)**

Token that indicates where to resume retrieving results on the next call to this action. If no token is returned, these results represent the end of the list.

Type: String


**Errors**

For information about the errors that are common to all actions, see Common Errors (p. 347).

**InternalServiceException**

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

**InvalidRequestException**

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

**UnsupportedRegionException**

The requested operation is not supported in the region specified.

HTTP Status Code: 400

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
DescribeMatchmakingRuleSets

Retrieves the details for FlexMatch matchmaking rule sets. You can request all existing rule sets for the region, or provide a list of one or more rule set names. When requesting multiple items, use the pagination parameters to retrieve results as a set of sequential pages. If successful, a rule set is returned for each requested name.

Operations related to match configurations and rule sets include:

- CreateMatchmakingConfiguration (p. 41)
- DescribeMatchmakingConfigurations (p. 134)
- UpdateMatchmakingConfiguration (p. 253)
- DeleteMatchmakingConfiguration (p. 75)
- CreateMatchmakingRuleSet (p. 46)
- DescribeMatchmakingRuleSets (p. 137)
- ValidateMatchmakingRuleSet (p. 262)

Request Syntax

```
{
  "Limit": number,
  "Names": [ "string" ],
  "NextToken": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.

- **Limit (p. 137)**
  
  Maximum number of results to return. Use this parameter with NextToken to get results as a set of sequential pages.

  Type: Integer


  Required: No

- **Names (p. 137)**

  Unique identifier for a matchmaking rule set. This name is used to identify the rule set associated with a matchmaking configuration.

  Type: Array of strings

  Array Members: Minimum number of 1 item. Maximum number of 10 items.

Response Syntax

```
{
   "NextToken": "string",
   "RuleSets": [
      {
         "CreationTime": number,
         "RuleSetBody": "string",
         "RuleSetName": "string"
      }
   ]
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**RuleSets (p. 138)**

Collection of requested matchmaking rule set objects.

Type: Array of MatchmakingRuleSet (p. 315) objects

**NextToken (p. 138)**

Token that indicates where to resume retrieving results on the next call to this action. If no token is returned, these results represent the end of the list.

Type: String


Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

**InternalServiceException**

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.
HTTP Status Code: 500
InvalidRequestException
One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400
NotFoundException
A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400
UnsupportedRegionException
The requested operation is not supported in the region specified.

HTTP Status Code: 400

See Also
For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
DescribePlayerSessions

Retrieves properties for one or more player sessions. This action can be used in several ways: (1) provide a PlayerSessionId to request properties for a specific player session; (2) provide a GameSessionId to request properties for all player sessions in the specified game session; (3) provide a PlayerId to request properties for all player sessions of a specified player.

To get game session record(s), specify only one of the following: a player session ID, a game session ID, or a player ID. You can filter this request by player session status. Use the pagination parameters to retrieve results as a set of sequential pages. If successful, a PlayerSession (p. 325) object is returned for each session matching the request.

Available in Amazon GameLift Local.

Player-session-related operations include:

- CreatePlayerSession (p. 49)
- CreatePlayerSessions (p. 52)
- DescribePlayerSessions (p. 140)
- Game session placements
  - StartGameSessionPlacement (p. 204)
  - DescribeGameSessionPlacement (p. 117)
  - StopGameSessionPlacement (p. 222)

Request Syntax

```json
{
  "GameSessionId": "string",
  "Limit": number,
  "NextToken": "string",
  "PlayerId": "string",
  "PlayerSessionId": "string",
  "PlayerSessionStatusFilter": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.

**GameSessionId (p. 140)**

Unique identifier for the game session to retrieve player sessions for.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: [a-zA-Z0-9:/-]+
Required: No

Limit (p. 140)

Maximum number of results to return. Use this parameter with NextToken to get results as a set of sequential pages. If a player session ID is specified, this parameter is ignored.

Type: Integer

Valid Range: Minimum value of 1.

Required: No

NextToken (p. 140)

Token that indicates the start of the next sequential page of results. Use the token that is returned with a previous call to this action. To start at the beginning of the result set, do not specify a value. If a player session ID is specified, this parameter is ignored.

Type: String


Required: No

PlayerId (p. 140)

Unique identifier for a player to retrieve player sessions for.

Type: String


Required: No

PlayerSessionId (p. 140)

Unique identifier for a player session to retrieve.

Type: String

Pattern: ^psess-\S+

Required: No

PlayerSessionStatusFilter (p. 140)

Player session status to filter results on.

Possible player session statuses include the following:

- **RESERVED** -- The player session request has been received, but the player has not yet connected to the server process and/or been validated.
- **ACTIVE** -- The player has been validated by the server process and is currently connected.
- **COMPLETED** -- The player connection has been dropped.
- **TIMEDOUT** -- A player session request was received, but the player did not connect and/or was not validated within the timeout limit (60 seconds).

Type: String


Required: No
Response Syntax

```
{
  "NextToken": "string",
  "PlayerSessions": [
  {
    "CreationTime": number,
    "FleetId": "string",
    "GameSessionId": "string",
    "IpAddress": "string",
    "PlayerData": "string",
    "PlayerId": "string",
    "PlayerSessionId": "string",
    "Port": number,
    "Status": "string",
    "TerminationTime": number
  }
  ]
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**NextToken (p. 142)**

Token that indicates where to resume retrieving results on the next call to this action. If no token is returned, these results represent the end of the list.

Type: String


**PlayerSessions (p. 142)**

Collection of objects containing properties for each player session that matches the request.

Type: Array of PlayerSession (p. 325) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

**InternalServiceException**

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

**InvalidRequestException**

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400
**NotFoundException**

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

**UnauthorizedException**

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
DescribeRuntimeConfiguration

Retrieves the current run-time configuration for the specified fleet. The run-time configuration tells Amazon GameLift how to launch server processes on instances in the fleet.

Fleet-related operations include:

- CreateFleet (p. 19)
- ListFleets (p. 173)
- DeleteFleet (p. 70)
- Describe fleets:
  - DescribeFleetAttributes (p. 93)
  - DescribeFleetCapacity (p. 97)
  - DescribeFleetPortSettings (p. 107)
  - DescribeFleetUtilization (p. 110)
  - DescribeRuntimeConfiguration (p. 144)
  - DescribeEC2InstanceLimits (p. 90)
  - DescribeFleetEvents (p. 101)
- Update fleets:
  - UpdateFleetAttributes (p. 234)
  - UpdateFleetCapacity (p. 238)
  - UpdateFleetPortSettings (p. 242)
  - UpdateRuntimeConfiguration (p. 258)
- Manage fleet actions:
  - StartFleetActions (p. 201)
  - StopFleetActions (p. 219)

Request Syntax

```json
{
  "FleetId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.

**FleetId (p. 144)**

Unique identifier for a fleet to get the run-time configuration for.

Type: String

Pattern: ^fleet-\S+$
Response Syntax

```json
{
    "RuntimeConfiguration": {
        "GameSessionActivationTimeoutSeconds": number,
        "MaxConcurrentGameSessionActivations": number,
        "ServerProcesses": [
            {
                "ConcurrentExecutions": number,
                "LaunchPath": "string",
                "Parameters": "string"
            }
        ]
    }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**RuntimeConfiguration (p. 145)**

Instructions describing how server processes should be launched and maintained on each instance in the fleet.

Type: RuntimeConfiguration (p. 331) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

**InternalServiceException**

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

**InvalidRequestException**

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

**NotFoundException**

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

**UnauthorizedException**

The client failed authentication. Clients should not retry such requests.
HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
DescribeScalingPolicies

Retrieves all scaling policies applied to a fleet.

To get a fleet's scaling policies, specify the fleet ID. You can filter this request by policy status, such as to retrieve only active scaling policies. Use the pagination parameters to retrieve results as a set of sequential pages. If successful, set of ScalingPolicy (p. 334) objects is returned for the fleet.

A fleet may have all of its scaling policies suspended (StopFleetActions (p. 219)). This action does not affect the status of the scaling policies, which remains ACTIVE. To see whether a fleet's scaling policies are in force or suspended, call DescribeFleetAttributes (p. 93) and check the stopped actions.

Operations related to fleet capacity scaling include:

- DescribeFleetCapacity (p. 97)
- UpdateFleetCapacity (p. 238)
- DescribeEC2InstanceLimits (p. 90)
- Manage scaling policies:
  - PutScalingPolicy (p. 178) (auto-scaling)
  - DescribeScalingPolicies (p. 147) (auto-scaling)
  - DeleteScalingPolicy (p. 77) (auto-scaling)
- Manage fleet actions:
  - StartFleetActions (p. 201)
  - StopFleetActions (p. 219)

Request Syntax

```
{
  "FleetId": "string",
  "Limit": number,
  "NextToken": "string",
  "StatusFilter": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.

**FleetId (p. 147)**

Unique identifier for a fleet to retrieve scaling policies for.

Type: String

Pattern: ^fleet-\S+

Required: Yes
Limit (p. 147)
Maximum number of results to return. Use this parameter with NextToken to get results as a set of sequential pages.
Type: Integer
Valid Range: Minimum value of 1.
Required: No

NextToken (p. 147)
Token that indicates the start of the next sequential page of results. Use the token that is returned with a previous call to this action. To start at the beginning of the result set, do not specify a value.
Type: String
Required: No

StatusFilter (p. 147)
Scaling policy status to filter results on. A scaling policy is only in force when in an ACTIVE status.
• ACTIVE -- The scaling policy is currently in force.
• UPDATEREQUESTED -- A request to update the scaling policy has been received.
• UPDATING -- A change is being made to the scaling policy.
• DELETEREQUESTED -- A request to delete the scaling policy has been received.
• DELETING -- The scaling policy is being deleted.
• DELETED -- The scaling policy has been deleted.
• ERROR -- An error occurred in creating the policy. It should be removed and recreated.
Type: String
Valid Values: ACTIVE | UPDATE_REQUESTED | UPDATING | DELETE_REQUESTED | DELETING | DELETED | ERROR
Required: No

Response Syntax

```json
{
  "NextToken": "string",
  "ScalingPolicies": [
    {
      "ComparisonOperator": "string",
      "EvaluationPeriods": number,
      "FleetId": "string",
      "MetricName": "string",
      "Name": "string",
      "PolicyType": "string",
      "ScalingAdjustment": number,
      "ScalingAdjustmentType": "string",
      "Status": "string",
      "TargetConfiguration": {
        "TargetValue": number
      },
      "Threshold": number
    }
  ]
}
```
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

NextToken (p. 148)

Token that indicates where to resume retrieving results on the next call to this action. If no token is returned, these results represent the end of the list.

Type: String


ScalingPolicies (p. 148)

Collection of objects containing the scaling policies matching the request.

Type: Array of ScalingPolicy (p. 334) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

InternalServiceException

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

InvalidRequestException

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

NotFoundException

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

UnauthorizedException

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

Example

View active scaling policies

In this example, we want to get a list of active scaling policies for a specific fleet. We can do this by retrieving the list of all policies for the fleet filtered by status. The sample request illustrates the use of pagination parameters, Limit and NextToken, to retrieve multiple results in sequential sets.
HTTP requests are authenticated using an AWS Signature Version 4 signature in the Authorization header field.

Sample Request

```
POST / HTTP/1.1
Host: gamelift.us-west-2.amazonaws.com;
Accept-Encoding: identity
Content-Length: 336
User-Agent: aws-cli/1.11.36 Python/2.7.9 Windows/7 botocore/1.4.93
Authorization: AWS4-HMAC-SHA256  Credential=AKIAIOSFODNN7EXAMPLE/20170406/us-west-2/gamelift/aws4_request, SignedHeaders=content-type;host;x-amz-date;x-amz-target, Signature=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
X-Amz-Date: 20170406T004805Z
X-Amz-Target: GameLift.DescribeScalingPolicies

{  
    "FleetId": "fleet-2222bbbc-33cc-44dd-55ee-6666ffff77aa",  
    "Limit": "2",  
    "StatusFilter": "ACTIVE"  
}
```

CLI syntax:

```
$aws gamelift describe-scaling-policies
--fleet-id "fleet-2222bbbc-33cc-44dd-55ee-6666ffff77aa"
--limit "2"
--status-filter "ACTIVE"
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: b34f8665-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 600
Date: Thu, 06 Apr 2017 00:48:07 GMT

{
    "ScalingPolicies": [  
        {  
            "Status": "ACTIVE",  
            "Name": "My_Target_Policy_1",  
            "FleetId": "fleet-2222bbbc-33cc-44dd-55ee-6666ffff77aa",  
            "PolicyType": "TargetBased",  
            "MetricName": "PercentAvailableGameSessions",  
            "TargetConfiguration": {"TargetValue": 15}  
        },  
        {  
            "Status": "ACTIVE",  
            "EvaluationPeriods": "1",  
            "Name": "My_Rule_Policy_1",  
            "FleetId": "fleet-2222bbbc-33cc-44dd-55ee-6666ffff77aa",  
            "PolicyType": "RuleBased",  
            "Threshold": "1.0",  
            "ScalingAdjustment": "10",  
            "MetricName": "QueueDepth",  
            "ScalingAdjustmentType": "PercentChangeInCapacity"  
        }  
    ],
```

API Version 2015-10-01

150
See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
DescribeVpcPeeringAuthorizations

Retrieves valid VPC peering authorizations that are pending for the AWS account. This operation returns all VPC peering authorizations and requests for peering. This includes those initiated and received by this account.

VPC peering connection operations include:

- CreateVpcPeeringAuthorization (p. 56)
- DescribeVpcPeeringAuthorizations (p. 152)
- DeleteVpcPeeringAuthorization (p. 80)
- CreateVpcPeeringConnection (p. 61)
- DescribeVpcPeeringConnections (p. 154)
- DeleteVpcPeeringConnection (p. 82)

Response Syntax

```json
{
   "VpcPeeringAuthorizations": [
      {
         "CreationTime": number,
         "ExpirationTime": number,
         "GameLiftAwsAccountId": "string",
         "PeerVpcAwsAccountId": "string",
         "PeerVpcId": "string"
      }
   ]
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**VpcPeeringAuthorizations (p. 152)**

Collection of objects that describe all valid VPC peering operations for the current AWS account.

Type: Array of **VpcPeeringAuthorization (p. 340)** objects

Errors

For information about the errors that are common to all actions, see **Common Errors (p. 347)**.

**InternalServiceException**

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

**InvalidRequestException**

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.
HTTP Status Code: 400

UnauthorizedException

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
DescribeVpcPeeringConnections

Retrieves information on VPC peering connections. Use this operation to get peering information for all fleets or for one specific fleet ID.

To retrieve connection information, call this operation from the AWS account that is used to manage the Amazon GameLift fleets. Specify a fleet ID or leave the parameter empty to retrieve all connection records. If successful, the retrieved information includes both active and pending connections. Active connections identify the IPv4 CIDR block that the VPC uses to connect.

VPC peering connection operations include:

- CreateVpcPeeringAuthorization (p. 56)
- DescribeVpcPeeringAuthorizations (p. 152)
- DeleteVpcPeeringAuthorization (p. 80)
- CreateVpcPeeringConnection (p. 61)
- DescribeVpcPeeringConnections (p. 154)
- DeleteVpcPeeringConnection (p. 82)

Request Syntax

```json
{
   "FleetId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.

**FleetId (p. 154)**

Unique identifier for a fleet.

Type: String

Pattern: ^fleet-\S+

Required: No

Response Syntax

```json
{
   "VpcPeeringConnections": [
      {
         "FleetId": "string",
         "GameLiftVpcId": "string",
         "IpV4CidrBlock": "string",
         "Status": "string",
         "RequestId": "string",
         "CreatedTimestamp": "string",
         "UpdatedTimestamp": "string"
      }
   ]
}
```
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

VpcPeeringConnections (p. 154)

Collection of VPC peering connection records that match the request.

Type: Array of VpcPeeringConnection (p. 342) objects

Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

InternalServiceException

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

InvalidRequestException

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

NotFoundException

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

UnauthorizedException

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
See Also

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
GetGameSessionLogUrl

Retrieves the location of stored game session logs for a specified game session. When a game session is terminated, Amazon GameLift automatically stores the logs in Amazon S3 and retains them for 14 days. Use this URL to download the logs.

**Note**
See the AWS Service Limits page for maximum log file sizes. Log files that exceed this limit are not saved.

Game-session-related operations include:

- CreateGameSession (p. 31)
- DescribeGameSessions (p. 123)
- DescribeGameSessionDetails (p. 113)
- SearchGameSessions (p. 192)
- UpdateGameSession (p. 246)
- GetGameSessionLogUrl (p. 157)

- Game session placements
  - StartGameSessionPlacement (p. 204)
  - DescribeGameSessionPlacement (p. 117)
  - StopGameSessionPlacement (p. 222)

**Request Syntax**

```json
{
   "GameSessionId": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**
In the following list, the required parameters are described first.

**GameSessionId (p. 157)**

Unique identifier for the game session to get logs for.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: [a-zA-Z0-9:/-]+

Required: Yes

**Response Syntax**

```json
{
}
```
"PreSignedUrl": "string"
}

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

PreSignedUrl (p. 157)

Location of the requested game session logs, available for download. This URL is valid for 15 minutes, after which S3 will reject any download request using this URL. You can request a new URL any time within the 14-day period that the logs are retained.

Type: String

Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

InternalServiceException

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

InvalidRequestException

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

NotFoundException

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

UnauthorizedException

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
See Also

- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
GetInstanceAccess

Requests remote access to a fleet instance. Remote access is useful for debugging, gathering benchmarking data, or watching activity in real time.

Access requires credentials that match the operating system of the instance. For a Windows instance, Amazon GameLift returns a user name and password as strings for use with a Windows Remote Desktop client. For a Linux instance, Amazon GameLift returns a user name and RSA private key, also as strings, for use with an SSH client. The private key must be saved in the proper format to a .pem file before using. If you're making this request using the AWS CLI, saving the secret can be handled as part of the GetInstanceAccess request. (See the example later in this topic). For more information on remote access, see Remotely Accessing an Instance.

To request access to a specific instance, specify the IDs of both the instance and the fleet it belongs to. You can retrieve a fleet's instance IDs by calling DescribeInstances (p. 127). If successful, an InstanceAccess (p. 306) object is returned containing the instance's IP address and a set of credentials.

Request Syntax

```
{
    "FleetId": "string",
    "InstanceId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.

**FleetId (p. 160)**

Unique identifier for a fleet that contains the instance you want access to. The fleet can be in any of the following statuses: ACTIVATING, ACTIVE, or ERROR. Fleets with an ERROR status may be accessible for a short time before they are deleted.

Type: String

Pattern: ^fleet-\S+

Required: Yes

**InstanceId (p. 160)**

Unique identifier for an instance you want to get access to. You can access an instance in any status.

Type: String

Pattern: [a-zA-Z0-9.-]+

Required: Yes
Response Syntax

```json
{
   "InstanceAccess": {
      "Credentials": {
         "Secret": "string",
         "UserName": "string"
      },
      "FleetId": "string",
      "InstanceId": "string",
      "IpAddress": "string",
      "OperatingSystem": "string"
   }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**InstanceAccess (p. 161)**

Object that contains connection information for a fleet instance, including IP address and access credentials.

Type: InstanceAccess (p. 306) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

**InternalServiceException**

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

**InvalidRequestException**

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

**NotFoundException**

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

**UnauthorizedException**

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400
Examples

Get credentials for a Linux instance

This example requests a set of credentials to remotely connect to a fleet instance running Linux.

If you're calling GetInstanceAccess programmatically, as with the JSON syntax, you need to save the returned value of Secret (an RSA private key) as a .pem file in the proper format. The returned value uses a newline (\n) to indicate a line break.

If you're calling GetInstanceAccess with the AWS CLI, you can automatically store the RSA private key as a .pem file in the proper format. See the CLI syntax example, which saves the private key to a file called MyPrivateKey.pem. Once the private key is saved, update the file permissions with the following command:

```
$ chmod 400 MyPrivateKey.pem
```

HTTP requests are authenticated using an AWS Signature Version 4 signature in the Authorization header field.

Sample Request

```plaintext
POST / HTTP/1.1
Host: gamelift.us-west-2.amazonaws.com;
Accept-Encoding: identity
Content-Length: 105
User-Agent: aws-cli/1.11.36 Python/2.7.9 Windows/7 botocore/1.4.93
Content-Type: application/x-amz-json-1.0
Authorization: AWS4-HMAC-SHA256  Credential=AKIAIOSFODNN7EXAMPLE/20170406/us-west-2/gamelift/aws4_request, SignedHeaders=content-type;host;x-amz-date;x-amz-target, Signature=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
X-Amz-Date: 20170406T004805Z
X-Amz-Target: GameLift.GetInstanceAccess

{"FleetId": "fleet-9999ffff-88ee-77dd-66cc-5555bbbb44aa",
 "InstanceId": "i-11111111a222b333c"
}
```

CLI syntax:
```
aws gamelift get-instance-access --fleet-id "fleet-9999ffff-88ee-77dd-66cc-5555bbbb44aa" --instance-id "i-11111111a222b333c" --query 'InstanceAccess.Credentials.Secret' --output text > MyPrivateKey.pem
```

Sample Response

```plaintext
HTTP/1.1 200 OK
x-amzn-RequestId: b34f8665-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 1986
Date: Thu, 06 Apr 2017 00:48:07 GMT

{""InstanceAccess": {
 ""Credentials": {
 ""Secret": "-----BEGIN RSA PRIVATE KEY-----
 nEXAMPLEKEYEYCAQEpAY7W2hasQrA1W3mRlQt6wyORRX8gngxgDAfRt/gx42kWXeT4rXE/b5CpSgie/
 \nvBoU7jLxx2pNHoFbY+P+Dc2l4eyz6CvjrhmAwOJwFwWi5/a/kH7iO5dsrvC7dqK2wUv5QbU6E0QW
 \n3/a/NXMr1GQEQX3gFWOxXVbwrerQo+ZWqeqiUwwMKuEBLeeJFLhMCVYURpUMeC1oehm449x9x1F
 \n50TCFeOzfl8dqCp6Gb2bP4jU195X/a/zORr9V+tpU0ZEL+w3XnZt3/nHPQxvxV2OJH67km6SUFP
 \nnoPzKe/D6V4x+MhHeF5r67DVOFjFBVWxKgBdZCRU2/weisD/HYwIDQAoABAoI1Q2ZaEvruru
 \n/n/uler7vIn5m7NSLKW4hJALIhU6UT/fzvctHK05kbcQXurlHMgQ2McyJX/0kn2NFjLV/
-----END RSA PRIVATE KEY-----"

API Version 2015-10-01
162
```
Amazon GameLift Service API Reference
Examples
ufGxbL1\nmb5qwMGUnEpJaZD6QSSs3kICLwWUYUiGfc0uiSbmJoap/
GTLU0W5Mfcv36PaBUNy5p53V6G7hXb2\nbahyWyJNfjLe4M86yd2YK3V2CmK+X/
BOsShnJ36+hjrXPPWmV3N9zEmCdJjA+K15DYmhm/
tJWSD9\n81oGk9TopEp7CkIfatEATyyZiVqoRq6k64iuM9JkA3OzdXzMQexXVJ1TLZVEH0E7bhlY9d8O1ozR
\noQs/FiZNAx2iijCWyv0lpjE73+kCgYEA9mZtyhkHkFDpwrSM1APaL8oNAbbjwEy7Z5Mqfql
+lIp1\nYkriL0DbLXlvRAH+yHPRit2hHOjtUNZh4Axv+cpg09qbUI3+43eEy24B7G/Uh+GTfbjsXsOxQx/
x\np9otyVwc7hsQ5TA5PZb+mvkJ5OBEKzet9XcKwONBYELGhnEPe7cCgYEA06Vgov6YHleHui9kHuws
\nayav0elc5zkxjF9nfHFJRry21R1trw2Vdpn+9g481URrpzWVOEihvm+xTtmaZlSp//lkq75XDwnU
\nWA8gkn6O3QE3fq2yN98BURsAKdJfJ5RL1HvGQvTe10HLYYXpJnEkHv+Unl2ajLivWUt5pbBrKbUC
\ngYBjbO+OZk0sCcpZ29sbzjYjpIddErySIyRX5gV2uNQwAjLdp9PfN295yQ+BxMBXiIycWVQiw0bH
\noMo7yykABY7Ozd5wQewBQ4AdSlWSX4nGDtsiFxWiI5sKuAAeOCbTosy1s8w8fxoJ5Tz1sdoxNeGs
\nArq6Wv/G16zQuAE9zK9vvwKBgF+09VI/1wJBirsDGz9whVWfFPrTkJNvJZzYt69qezxlsjgFKshy
\nWBhd4xHZtmCqpBPlAymEjr/TOlbxyARmXMnIOWIAnNXMGB4KGSyl1mzSVAoQ+fqR+cJ3d0dyPl1j
\njjb0Ed/NY8frlNDxAVHE8BSkdsx2f6ELEyBKJSRr9snRAoGAMrTwYneXzvTskF/S5Fyu0iOegLDa
\nNWUH38v/nDCgEpIXD5Hn3qAEcju1IjmbwlvtW+nY2jVhv7UGd8MjwUTNGItdb6nsYqM2asrnF3qS
\nVRkAKKKYeGjkpUfVTrW0YFjXkfcrR/V+QFL5OndHAKJXjW7a4ejJLncTzmZSpYzwApc=\n-----END RSA
PRIVATE KEY-----",
"UserName": "gl-user-remote"
},
"FleetId": "fleet-9999ffff-88ee-77dd-66cc-5555bbbb44aa",
"InstanceId": "i-11111111a222b333c",
"IpAddress": "192.0.2.0",
"OperatingSystem": "AMAZON_LINUX"
}

Get credentials for a Windows instance
This example requests a set of credentials to remotely connect to a ﬂeet instance running Windows.
HTTP requests are authenticated using an AWS Signature Version 4 signature in the Authorization header
ﬁeld.

Sample Request
POST / HTTP/1.1
Host: gamelift.us-west-2.amazonaws.com;
Accept-Encoding: identity
Content-Length: 105
User-Agent: aws-cli/1.11.36 Python/2.7.9 Windows/7 botocore/1.4.93
Content-Type: application/x-amz-json-1.0
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20170406/us-west-2/
gamelift/aws4_request, SignedHeaders=content-type;host;x-amz-date;x-amz-target,
Signature=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
X-Amz-Date: 20170406T004805Z
X-Amz-Target: GameLift.GetInstanceAccess
{"FleetId": "fleet-9999ffff-88ee-77dd-66cc-5555bbbb44aa",
"InstanceId": "i-11111111a222b333c"
}
CLI syntax:
aws gamelift get-instance-access --fleet-id
"fleet-9999ffff-88ee-77dd-66cc-5555bbbb44aa" --instance-id "i-11111111a222b333c"

Sample Response
HTTP/1.1 200 OK
x-amzn-RequestId: b34f8665-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 1986
Date: Thu, 06 Apr 2017 00:48:07 GMT

API Version 2015-10-01
163


"InstanceAccess": {
  "Credentials": {
    "Secret": "aA1bBB2cCCd3EEE",
    "UserName": "gl-user-remote"
  },
  "FleetId": "fleet-9999ffff-88ee-77dd-66cc-5555bbbb44aa",
  "InstanceId": "i-11111111a222b333c",
  "IpAddress": "192.0.2.0",
  "OperatingSystem": "WIN_2012"
}

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
ListAliases

Retrieves all aliases for this AWS account. You can filter the result set by alias name and/or routing strategy type. Use the pagination parameters to retrieve results in sequential pages.

Note
Returned aliases are not listed in any particular order.

Alias-related operations include:

- CreateAlias (p. 10)
- ListAliases (p. 165)
- DescribeAlias (p. 84)
- UpdateAlias (p. 227)
- DeleteAlias (p. 65)
- ResolveAlias (p. 189)

Request Syntax

```
{
  "Limit": number,
  "Name": "string",
  "NextToken": "string",
  "RoutingStrategyType": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

Note
In the following list, the required parameters are described first.

**Limit (p. 165)**

Maximum number of results to return. Use this parameter with `NextToken` to get results as a set of sequential pages.

Type: Integer

Valid Range: Minimum value of 1.

Required: No

**Name (p. 165)**

Descriptive label that is associated with an alias. Alias names do not need to be unique.

Type: String

Length Constraints: Minimum length of 1.

Required: No
**NextToken (p. 165)**

Token that indicates the start of the next sequential page of results. Use the token that is returned with a previous call to this action. To start at the beginning of the result set, do not specify a value.

Type: String

Length Constraints: Minimum length of 1.

Required: No

**RoutingStrategyType (p. 165)**

Type of routing to filter results on. Use this parameter to retrieve only aliases of a certain type. To retrieve all aliases, leave this parameter empty.

Possible routing types include the following:
- **SIMPLE** -- The alias resolves to one specific fleet. Use this type when routing to active fleets.
- **TERMINAL** -- The alias does not resolve to a fleet but instead can be used to display a message to the user. A terminal alias throws a TerminalRoutingStrategyException with the RoutingStrategy (p. 329) message embedded.

Type: String

Valid Values: SIMPLE | TERMINAL

Required: No

---

**Response Syntax**

```json
{
   "Aliases": [
      {
         "AliasArn": "string",
         "AliasId": "string",
         "CreationTime": number,
         "Description": "string",
         "LastUpdatedTime": number,
         "Name": "string",
         "RoutingStrategy": {
            "FleetId": "string",
            "Message": "string",
            "Type": "string"
         }
      }
   ],
   "NextToken": "string"
}
```

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**Aliases (p. 166)**

Collection of alias records that match the list request.
Type: Array of Alias (p. 266) objects

NextToken (p. 166)

Token that indicates where to resume retrieving results on the next call to this action. If no token is returned, these results represent the end of the list.

Type: String

Length Constraints: Minimum length of 1.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

InternalServiceException

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

InvalidRequestException

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

UnauthorizedException

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
ListBuilds

Retrieves build records for all builds associated with the AWS account in use. You can limit results to builds that are in a specific status by using the `Status` parameter. Use the pagination parameters to retrieve results in a set of sequential pages.

**Note**
Build records are not listed in any particular order.

Build-related operations include:

- `CreateBuild` (p. 13)
- `ListBuilds` (p. 168)
- `DescribeBuild` (p. 87)
- `UpdateBuild` (p. 230)
- `DeleteBuild` (p. 67)

**Request Syntax**

```
{
  "Limit": number,
  "NextToken": "string",
  "Status": "string"
}
```

**Request Parameters**

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**
In the following list, the required parameters are described first.

**Limit** (p. 168)

Maximum number of results to return. Use this parameter with `NextToken` to get results as a set of sequential pages.

Type: Integer

Valid Range: Minimum value of 1.

Required: No

**NextToken** (p. 168)

Token that indicates the start of the next sequential page of results. Use the token that is returned with a previous call to this action. To start at the beginning of the result set, do not specify a value.

Type: String

Length Constraints: Minimum length of 1.
Required: No

**Status (p. 168)**

Build status to filter results by. To retrieve all builds, leave this parameter empty.

Possible build statuses include the following:

- **INITIALIZED** -- A new build has been defined, but no files have been uploaded. You cannot create fleets for builds that are in this status. When a build is successfully created, the build status is set to this value.

- **READY** -- The game build has been successfully uploaded. You can now create new fleets for this build.

- **FAILED** -- The game build upload failed. You cannot create new fleets for this build.

Type: String

Valid Values: INITIALIZED | READY | FAILED

Required: No

**Response Syntax**

```
{
  "Builds": [
    {
      "BuildId": "string",
      "CreationTime": number,
      "Name": "string",
      "OperatingSystem": "string",
      "SizeOnDisk": number,
      "Status": "string",
      "Version": "string"
    }
  ],
  "NextToken": "string"
}
```

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**Builds (p. 169)**

Collection of build records that match the request.

Type: Array of Build (p. 271) objects

**NextToken (p. 169)**

Token that indicates where to resume retrieving results on the next call to this action. If no token is returned, these results represent the end of the list.

Type: String

Length Constraints: Minimum length of 1.
Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

**InternalServiceException**

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

**InvalidRequestException**

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

**UnauthorizedException**

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

Examples

Retrieve all builds

This example retrieves game server builds in the current region. The sample request illustrates how to use the pagination parameters, `Limit` and `NextToken`, to retrieve the results in sequential sets.


**Sample Request**

```plaintext
POST / HTTP/1.1
Host: gamelift.us-west-2.amazonaws.com
Accept-Encoding: identity
Content-Length: 171
User-Agent: aws-cli/1.11.36 Python/2.7.9 Windows/7 botocore/1.4.93
Content-Type: application/x-amz-json-1.0
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20170406/us-west-2/gamelift/aws4_request, SignedHeaders=content-type;host;x-amz-date;x-amz-target, Signature=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
X-Amz-Date: 20170406T004805Z
X-Amz-Target: GameLift.ListBuilds
{
    "Limit": 2,
    "NextToken": "eyJhd3NBY2NvdW50SWQiOnsicyI6IjMwMjc3NjAxNjM5OCJ9LCJidWlsZElkIjp7InMiOiJidWlsZC00NDRlZjQxZS1hM2I1LTQ2NDYtODJmMy0zYzI4ZTgxNjVjEXAMPLE="
}
```

**Sample Response**

```
HTTP/1.1 200 OK
x-amzn-RequestId: b34f8665-EXAMPLE
Content-Type: application/x-amz-json-1.1
```
Retrieve failed builds

This example retrieves all game server builds in the current region that failed to upload. It uses the pagination parameters to retrieve two builds at a time. With no NextToken provided, this request tries to retrieve the first two results. There is only one result, so only one build record and no NextToken is returned.

HTTP requests are authenticated using an AWS Signature Version 4 signature in the Authorization header field.

Sample Request

POST / HTTP/1.1
Host: gamelift.us-west-2.amazonaws.com;
Accept-Encoding: identity
Content-Length: 42
User-Agent: aws-cli/1.11.36 Python/2.7.9 Windows/7 botocore/1.4.93
Content-Type: application/x-amz-json-1.0
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20170406/us-west-2/gamelift/aws4_request, SignedHeaders=content-type;host;x-amz-date;x-amz-target, Signature=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
X-Amz-Date: 20170406T004805Z
X-Amz-Target: GameLift.ListBuilds

{
   "Limit": 2,
   "Status": "FAILED"
}

Sample Response

HTTP/1.1 200 OK
x-amzn-RequestId: b34f8665-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 351
See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
ListFleets

Retrieves a collection of fleet records for this AWS account. You can filter the result set by build ID. Use the pagination parameters to retrieve results in sequential pages.

Note
Fleet records are not listed in any particular order.

Fleet-related operations include:

- CreateFleet (p. 19)
- ListFleets (p. 173)
- DeleteFleet (p. 70)
- Describe fleets:
  - DescribeFleetAttributes (p. 93)
  - DescribeFleetCapacity (p. 97)
  - DescribeFleetPortSettings (p. 107)
  - DescribeFleetUtilization (p. 110)
  - DescribeRuntimeConfiguration (p. 144)
  - DescribeEC2InstanceLimits (p. 90)
  - DescribeFleetEvents (p. 101)
- Update fleets:
  - UpdateFleetAttributes (p. 234)
  - UpdateFleetCapacity (p. 238)
  - UpdateFleetPortSettings (p. 242)
  - UpdateRuntimeConfiguration (p. 258)
- Manage fleet actions:
  - StartFleetActions (p. 201)
  - StopFleetActions (p. 219)

Request Syntax

```json
{
  "BuildId": "string",
  "Limit": number,
  "NextToken": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

Note
In the following list, the required parameters are described first.

BuildId (p. 173)

Unique identifier for a build to return fleets for. Use this parameter to return only fleets using the specified build. To retrieve all fleets, leave this parameter empty.
Type: String

Pattern: ^build-\S+

Required: No

Limit (p. 173)

Maximum number of results to return. Use this parameter with NextToken to get results as a set of sequential pages.

Type: Integer

Valid Range: Minimum value of 1.

Required: No

NextToken (p. 173)

Token that indicates the start of the next sequential page of results. Use the token that is returned with a previous call to this action. To start at the beginning of the result set, do not specify a value.

Type: String


Required: No

Response Syntax

```
{
    "FleetIds": [ "string" ],
    "NextToken": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

FleetIds (p. 174)

Set of fleet IDs matching the list request. You can retrieve additional information about all returned fleets by passing this result set to a call to DescribeFleetAttributes (p. 93), DescribeFleetCapacity (p. 97), or DescribeFleetUtilization (p. 110).

Type: Array of strings

Array Members: Minimum number of 1 item.

Pattern: ^fleet-\S+

NextToken (p. 174)

Token that indicates where to resume retrieving results on the next call to this action. If no token is returned, these results represent the end of the list.

Type: String
Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

InternalServiceException

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

InvalidRequestException

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

NotFoundException

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

UnauthorizedException

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

Examples

List all fleets in the region

This example retrieves the fleet IDs of all existing fleets, regardless of build and status, and uses the pagination parameters to retrieve two fleet IDs at a time. This request illustrates how to retrieve a subsequent set of results by using a NextToken token that was returned with a previous request. The example response also includes a NextToken, which indicates that there are still more results to retrieve. To get more information on an individual fleet, use the Describe Fleet operations with a fleet ID.

HTTP requests are authenticated using an AWS Signature Version 4 signature in the Authorization header field.

Sample Request

```
POST / HTTP/1.1
Host: gamelift.us-west-2.amazonaws.com;
Accept-Encoding: identity
Content-Length: 208
User-Agent: aws-cli/1.11.36 Python/2.7.9 Windows/7 botocore/1.4.93
Content-Type: application/x-amz-json-1.0
Authorization: AWS4-HMAC-SHA256  Credential=AKIAIOSFODNN7EXAMPLE/20170406/us-west-2/
gamelift/aws4_request, SignedHeaders=content-type;host;x-amz-date;x-amz-target,
Signature=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
X-Amz-Date: 20170406T004805Z
X-Amz-Target: GameLift.ListFleets
```
Sample Response

HTTP/1.1 200 OK
x-amzn-RequestID: b34f8665-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 264
Date: Thu, 06 Apr 2017 00:48:07 GMT

{
  "FleetIds": [
    "fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
    "fleet-9999ffff-88ee-77dd-66cc-5555bbbb44aa"
  ],
  "NextToken":
    "eyJhd3NBY2NvdW50SWQiOnsicyI6IjMwMjc3NjAxNjM5OCJ9LCJidWlsZElkIjp7InMiOiJidWlsZC01NWYxZTZmMS1jY2FlLTQ3YTctOWI5ZS1iYjFkYTQwMjEXAMPLE2"
}

List all fleets in the region for a specific build

This example retrieves the IDs of fleets that are deployed with a specified game build, regardless of status. This request does not use the pagination parameters. Without a limit, up to 16 results may be returned with any request. In this case, there is only one fleet. To get more information on an individual fleet, call the DescribeFleet operations with the fleet ID.

HTTP requests are authenticated using an AWS Signature Version 4 signature in the Authorization header field.

Sample Request

POST / HTTP/1.1
Host: gamelift.us-west-2.amazonaws.com;
Accept-Encoding: identity
Content-Length: 63
User-Agent: aws-cli/1.11.36 Python/2.7.9 Windows/7 botocore/1.4.93
Content-Type: application/x-amz-json-1.0
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20170406/us-west-2/gamelift/aws4_request, SignedHeaders=content-type;host;x-amz-date;x-amz-target,
Signature=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
X-Amz-Date: 20170406T004805Z
X-Amz-Target: GameLift.ListFleets

{
  "Build": "build-1111aaaa-22bb-33cc-44dd-5555ee66ff"
}

Sample Response

HTTP/1.1 200 OK
x-amzn-RequestID: b34f8665-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 66
Date: Thu, 06 Apr 2017 00:48:07 GMT

{
"FleetIds": ["fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa"]
}

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
PutScalingPolicy

Creates or updates a scaling policy for a fleet. Scaling policies are used to automatically scale a fleet's hosting capacity to meet player demand. An active scaling policy instructs Amazon GameLift to track a fleet metric and automatically change the fleet's capacity when a certain threshold is reached. There are two types of scaling policies: target-based and rule-based. Use a target-based policy to quickly and efficiently manage fleet scaling; this option is the most commonly used. Use rule-based policies when you need to exert fine-grained control over auto-scaling.

Fleets can have multiple scaling policies of each type in force at the same time; you can have one target-based policy, one or multiple rule-based scaling policies, or both. We recommend caution, however, because multiple auto-scaling policies can have unintended consequences.

You can temporarily suspend all scaling policies for a fleet by calling StopFleetActions (p. 219) with the fleet action AUTO_SCALING. To resume scaling policies, call StartFleetActions (p. 201) with the same fleet action. To stop just one scaling policy--or to permanently remove it, you must delete the policy with DeleteScalingPolicy (p. 77).

Learn more about how to work with auto-scaling in Set Up Fleet Automatic Scaling.

Target-based policy

A target-based policy tracks a single metric: PercentAvailableGameSessions. This metric tells us how much of a fleet's hosting capacity is ready to host game sessions but is not currently in use. This is the fleet's buffer; it measures the additional player demand that the fleet could handle at current capacity. With a target-based policy, you set your ideal buffer size and leave it to Amazon GameLift to take whatever action is needed to maintain that target.

For example, you might choose to maintain a 10% buffer for a fleet that has the capacity to host 100 simultaneous game sessions. This policy tells Amazon GameLift to take action whenever the fleet's available capacity falls below or rises above 10 game sessions. Amazon GameLift will start new instances or stop unused instances in order to return to the 10% buffer.

To create or update a target-based policy, specify a fleet ID and name, and set the policy type to "TargetBased". Specify the metric to track (PercentAvailableGameSessions) and reference a TargetConfiguration (p. 339) object with your desired buffer value. Exclude all other parameters. On a successful request, the policy name is returned. The scaling policy is automatically in force as soon as it's successfully created. If the fleet's auto-scaling actions are temporarily suspended, the new policy will be in force once the fleet actions are restarted.

Rule-based policy

A rule-based policy tracks specified fleet metric, sets a threshold value, and specifies the type of action to initiate when triggered. With a rule-based policy, you can select from several available fleet metrics. Each policy specifies whether to scale up or scale down (and by how much), so you need one policy for each type of action.

For example, a policy may make the following statement: "If the percentage of idle instances is greater than 20% for more than 15 minutes, then reduce the fleet capacity by 10%.

A policy's rule statement has the following structure:

If [MetricName] is [ComparisonOperator] [Threshold] for [EvaluationPeriods] minutes, then [ScalingAdjustmentType]to/by [ScalingAdjustment].

To implement the example, the rule statement would look like this:

If [PercentIdleInstances] is [GreaterThanThreshold] [20] for [15] minutes, then [PercentChangeInCapacity]to/by [10].

To create or update a scaling policy, specify a unique combination of name and fleet ID, and set the policy type to "RuleBased". Specify the parameter values for a policy rule statement. On a successful request, the policy name is returned. Scaling policies are automatically in force as soon as they're successfully created. If the fleet's auto-scaling actions are temporarily suspended, the new policy will be in force once the fleet actions are restarted.

Operations related to fleet capacity scaling include:

- DescribeFleetCapacity (p. 97)
- UpdateFleetCapacity (p. 238)
- DescribeEC2InstanceLimits (p. 90)
- Manage scaling policies:
  - PutScalingPolicy (p. 178) (auto-scaling)
  - DescribeScalingPolicies (p. 147) (auto-scaling)
  - DeleteScalingPolicy (p. 77) (auto-scaling)
- Manage fleet actions:
  - StartFleetActions (p. 201)
  - StopFleetActions (p. 219)

### Request Syntax

```
{
    "ComparisonOperator": "string",
    "EvaluationPeriods": number,
    "FleetId": "string",
    "MetricName": "string",
    "Name": "string",
    "PolicyType": "string",
    "ScalingAdjustment": number,
    "ScalingAdjustmentType": "string",
    "TargetConfiguration": {
        "TargetValue": number
    },
    "Threshold": number
}
```

### Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.

**FleetId (p. 179)**

Unique identifier for a fleet to apply this policy to. The fleet cannot be in any of the following statuses: ERROR or DELETING.

Type: String

Pattern: ^fleet-\S+

Required: Yes
MetricName (p. 179)

Name of the Amazon GameLift-defined metric that is used to trigger a scaling adjustment. For detailed descriptions of fleet metrics, see Monitor Amazon GameLift with Amazon CloudWatch.

- **ActivatingGameSessions** -- Game sessions in the process of being created.
- **ActiveGameSessions** -- Game sessions that are currently running.
- **ActiveInstances** -- Fleet instances that are currently running at least one game session.
- **AvailableGameSessions** -- Additional game sessions that fleet could host simultaneously, given current capacity.
- **AvailablePlayerSessions** -- Empty player slots in currently active game sessions. This includes game sessions that are not currently accepting players. Reserved player slots are not included.
- **CurrentPlayerSessions** -- Player slots in active game sessions that are being used by a player or are reserved for a player.
- **IdleInstances** -- Active instances that are currently hosting zero game sessions.
- **PercentAvailableGameSessions** -- Unused percentage of the total number of game sessions that a fleet could host simultaneously, given current capacity. Use this metric for a target-based scaling policy.
- **PercentIdleInstances** -- Percentage of the total number of active instances that are hosting zero game sessions.
- **QueueDepth** -- Pending game session placement requests, in any queue, where the current fleet is the top-priority destination.
- **WaitTime** -- Current wait time for pending game session placement requests, in any queue, where the current fleet is the top-priority destination.

Type: String

Valid Values: ActivatingGameSessions | ActiveGameSessions | ActiveInstances | AvailableGameSessions | AvailablePlayerSessions | CurrentPlayerSessions | IdleInstances | PercentAvailableGameSessions | PercentIdleInstances | QueueDepth | WaitTime

Required: Yes

Name (p. 179)

Descriptive label that is associated with a scaling policy. Policy names do not need to be unique. A fleet can have only one scaling policy with the same name.

Type: String


Required: Yes

ComparisonOperator (p. 179)

Comparison operator to use when measuring the metric against the threshold value.

Type: String

Valid Values: GreaterThanOrEqualToThreshold | GreaterThanThreshold | LessThanThreshold | LessThanOrEqualToThreshold

Required: No

EvaluationPeriods (p. 179)

Length of time (in minutes) the metric must be at or beyond the threshold before a scaling event is triggered.
PolicyType (p. 179)

Type of scaling policy to create. For a target-based policy, set the parameter MetricName to 'PercentAvailableGameSessions' and specify a TargetConfiguration. For a rule-based policy set the following parameters: MetricName, ComparisonOperator, Threshold, EvaluationPeriods, ScalingAdjustmentType, and ScalingAdjustment.

Type: String
Valid Values: RuleBased | TargetBased
Required: No

ScalingAdjustment (p. 179)

Amount of adjustment to make, based on the scaling adjustment type.

Type: Integer
Required: No

ScalingAdjustmentType (p. 179)

Type of adjustment to make to a fleet's instance count (see FleetCapacity (p. 285)):

- **ChangeInCapacity** -- add (or subtract) the scaling adjustment value from the current instance count. Positive values scale up while negative values scale down.
- **ExactCapacity** -- set the instance count to the scaling adjustment value.
- **PercentChangeInCapacity** -- increase or reduce the current instance count by the scaling adjustment, read as a percentage. Positive values scale up while negative values scale down; for example, a value of "-10" scales the fleet down by 10%.

Type: String
Valid Values: ChangeInCapacity | ExactCapacity | PercentChangeInCapacity
Required: No

TargetConfiguration (p. 179)

Object that contains settings for a target-based scaling policy.

Type: TargetConfiguration (p. 339) object
Required: No

Threshold (p. 179)

Metric value used to trigger a scaling event.

Type: Double
Required: No

Response Syntax

```
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

Name (p. 181)

Descriptive label that is associated with a scaling policy. Policy names do not need to be unique.

Type: String


Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

InternalServerError

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

InvalidRequestException

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

NotFoundException

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

UnauthorizedException

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

Examples

Create a target-based scaling policy

This example sets up auto-scaling using a target-based scaling policy. For this fleet, we want to maintain a 15% capacity buffer for our game, so that our fleet will always be able to immediately accommodate some additional game sessions. For a target-based policy, we need to specify a fleet ID, policy name and type, metric name (set this parameter to "PercentAvailableGameSessions"), and target configuration (buffer size). Verify that the new policy has gone into effect by calling DescribeFleetAttributes (p. 93) to
check that auto-scaling actions for the fleet have not been stopped. Call DescribeScalingPolicies (p. 147) to view the newly created policy.

HTTP requests are authenticated using an AWS Signature Version 4 signature in the Authorization header field.

Sample Request

```
POST / HTTP/1.1
Host: gamelift.us-west-2.amazonaws.com;
Accept-Encoding: identity
Content-Length: 338
Content-Type: application/x-amz-json-1.0
Authorization: AWS4-HMAC-SHA256
   Credential=AKIAIOSFODNN7EXAMPLE/20170406/us-west-2/gamelift/aws4_request,
   SignedHeaders=content-type;host;x-amz-date;x-amz-target,
   Signature=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
X-Amz-Date: 20170406T004805Z
X-Amz-Target: GameLift.PutScalingPolicy
{
   "FleetId": "fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
   "Name": "My_Target_Policy_1",
   "PolicyType": "TargetBased",
   "MetricName": "PercentAvailableGameSessions",
   "TargetConfiguration": {
      "TargetValue": 15
   }
}
```

CLI syntax:

```
$aws gamelift put-scaling-policy
   --fleet-id "fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa"
   --name "My_Target_Policy_1"
   --policy-type "TargetBased"
   --metric-name "PercentAvailableGameSessions"
   --target-configuration "TargetValue=5"
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: b34f8665-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 607
Date: Thu, 06 Apr 2017 00:48:07 GMT
{
   "Name": "My_Target_Policy_1"
}
```

Create a rule-based scaling policy

This example illustrates using a rule-based policy to supplement a target-based policy. While the target policy does most of the work of ensuring that capacity tracks with player demand, a well-formed rule-based policy can handle special circumstances and edge cases. For example, the target-based approach becomes less efficient when fleets have just few instances. We can mitigate this issue by creating a rule that maintains at least one idle instance ready to host new game sessions. At low capacity, the two policies do not conflict; at higher capacity, the rule-based policy loses relevance.

HTTP requests are authenticated using an AWS Signature Version 4 signature in the Authorization header field.
Sample Request

```plaintext
POST / HTTP/1.1
Host: gamelift.us-west-2.amazonaws.com;
Accept-Encoding: identity
Content-Length: 336
User-Agent: aws-cli/1.11.36 Python/2.7.9 Windows/7 botocore/1.4.93
Authorization: AWS4-HMAC-SHA256  Credential=AKIAIOSFODNN7EXAMPLE/20170406/us-west-2/
gamelift/aws4_request, SignedHeaders=content-type;host;x-amz-date;x-amz-target,
Signature=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
X-Amz-Date: 20170406T004805Z
X-Amz-Target: GameLift.PutScalingPolicy
{
    "FleetId": "fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
    "Name": "My_Rule_Policy_1",
    "PolicyType": "RuleBased",
    "MetricName": "IdleInstances",
    "ComparisonOperator": "LessThanThreshold",
    "Threshold": "2",
    "EvaluationPeriods": "5",
    "ScalingAdjustmentType": "ChangeInCapacity",
    "ScalingAdjustment": "1"
}
```

CLI syntax:

```
$aws gamelift put-scaling-policy
--fleet-id "fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa"
--name "My_Rule_Policy_1"
--policy-type "RuleBased"
--metric-name "IdleInstances"
--comparison-operator "LessThanThreshold"
--threshold "2"
--evaluation-periods "5"
--scaling-adjustment-type "ChangeInCapacity"
--scaling-adjustment "1"
```

Sample Response

```
HTTP/1.1 200 OK
x-amzn-RequestId: b34f8665-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 600
Date: Thu, 06 Apr 2017 00:48:07 GMT
{
    "Name": "My_Rule_Policy_1"
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V2
RequestUploadCredentials

Retrieves a fresh set of credentials for use when uploading a new set of game build files to Amazon GameLift’s Amazon S3. This is done as part of the build creation process; see CreateBuild (p. 13).

To request new credentials, specify the build ID as returned with an initial CreateBuild request. If successful, a new set of credentials are returned, along with the S3 storage location associated with the build ID.

Request Syntax

```
{
    "BuildId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.

**BuildId (p. 186)**

Unique identifier for a build to get credentials for.

Type: String

Pattern: ^build-\S+

Required: Yes

Response Syntax

```
{
    "StorageLocation": {
        "Bucket": "string",
        "Key": "string",
        "RoleArn": "string"
    },
    "UploadCredentials": {
        "AccessKeyId": "string",
        "SecretAccessKey": "string",
        "SessionToken": "string"
    }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.
StorageLocation (p. 186)

Amazon S3 path and key, identifying where the game build files are stored.

Type: S3Location (p. 333) object

UploadCredentials (p. 186)

AWS credentials required when uploading a game build to the storage location. These credentials have a limited lifespan and are valid only for the build they were issued for.

Type: AwsCredentials (p. 270) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

InternalServiceException

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

InvalidRequestException

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

NotFoundException

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

UnauthorizedException

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

Example

Refresh access credentials for uploading a build

This example obtains new, valid access credentials for uploading a build file to Amazon GameLift’s S3 account. Credentials, which are provided when you create a build record using CreateBuild (p. 13), have a limited lifespan. The build ID required for this operation is returned in response to the original CreateBuild request.

HTTP requests are authenticated using an AWS Signature Version 4 signature in the Authorization header field.

Sample Request

POST / HTTP/1.1
Sample Response

HTTP/1.1 200 OK
x-amzn-RequestId: b34f8665-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 369
Date: Thu, 06 Apr 2017 00:48:07 GMT

{
  "StorageLocation": {
    "Bucket": "gamelift-builds-us-west-2",
    "Key": "123456789012/build-1111aaaa-22bb-33cc-44dd-5555eeee66ff"
  },
  "UploadCredentials": {
    "AccessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "SecretAccessKey": "wJalrXUtNEMI/K7MDENG/bPxrRfiCYEXAMPLEKEY",
    "SessionToken": "AgoGb3JpZ2luENz...EXAMPLETOKEN=="
  }
}

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
ResolveAlias

Retrieves the fleet ID that a specified alias is currently pointing to.

Alias-related operations include:

- CreateAlias (p. 10)
- ListAliases (p. 165)
- DescribeAlias (p. 84)
- UpdateAlias (p. 227)
- DeleteAlias (p. 65)
- ResolveAlias (p. 189)

Request Syntax

```
{
  "AliasId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.

**AliasId (p. 189)**

Unique identifier for the alias you want to resolve.

Type: String

Pattern: ^alias-\S+

Required: Yes

Response Syntax

```
{
  "FleetId": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.
FleetId (p. 189)

Fleet identifier that is associated with the requested alias.

Type: String

Pattern: ^fleet-\S+

Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

InternalServerError

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

InvalidRequestException

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

NotFoundException

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

TerminalRoutingStrategyException

The service is unable to resolve the routing for a particular alias because it has a terminal RoutingStrategy (p. 329) associated with it. The message returned in this exception is the message defined in the routing strategy itself. Such requests should only be retried if the routing strategy for the specified alias is modified.

HTTP Status Code: 400

UnauthorizedException

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V2
SearchGameSessions

Retrieves all active game sessions that match a set of search criteria and sorts them in a specified order. You can search or sort by the following game session attributes:

- **gameSessionId** -- Unique identifier for the game session. You can use either a GameSessionId or GameSessionArn value.
- **gameSessionName** -- Name assigned to a game session. This value is set when requesting a new game session with CreateGameSession (p. 31) or updating with UpdateGameSession (p. 246). Game session names do not need to be unique to a game session.
- **gameSessionProperties** -- Custom data defined in a game session's GameProperty parameter. GameProperty values are stored as key:value pairs; the filter expression must indicate the key and a string to search the data values for. For example, to search for game sessions with custom data containing the key:value pair "gameMode:brawl", specify the following:

```plaintext
gameSessionProperties.gameMode = "brawl"
```

All custom data values are searched as strings.
- **maximumSessions** -- Maximum number of player sessions allowed for a game session. This value is set when requesting a new game session with CreateGameSession (p. 31) or updating with UpdateGameSession (p. 246).
- **creationTimeMillis** -- Value indicating when a game session was created. It is expressed in Unix time as milliseconds.
- **playerSessionCount** -- Number of players currently connected to a game session. This value changes rapidly as players join the session or drop out.
- **hasAvailablePlayerSessions** -- Boolean value indicating whether a game session has reached its maximum number of players. It is highly recommended that all search requests include this filter attribute to optimize search performance and return only sessions that players can join.

**Note**

Returned values for playerSessionCount and hasAvailablePlayerSessions change quickly as players join sessions and others drop out. Results should be considered a snapshot in time. Be sure to refresh search results often, and handle sessions that fill up before a player can join.

To search or sort, specify either a fleet ID or an alias ID, and provide a search filter expression, a sort expression, or both. If successful, a collection of GameSession (p. 290) objects matching the request is returned. Use the pagination parameters to retrieve results as a set of sequential pages.

You can search for game sessions one fleet at a time only. To find game sessions across multiple fleets, you must search each fleet separately and combine the results. This search feature finds only game sessions that are in ACTIVE status. To locate games in statuses other than active, use DescribeGameSessionDetails (p. 113).

Game-session-related operations include:

- CreateGameSession (p. 31)
- DescribeGameSessions (p. 123)
- DescribeGameSessionDetails (p. 113)
- SearchGameSessions (p. 192)
- UpdateGameSession (p. 246)
- GetGameSessionLogUrl (p. 157)
- Game session placements
  - StartGameSessionPlacement (p. 204)
  - DescribeGameSessionPlacement (p. 117)
Request Syntax

```json
{
  "AliasId": "string",
  "FilterExpression": "string",
  "FleetId": "string",
  "Limit": number,
  "NextToken": "string",
  "SortExpression": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**
In the following list, the required parameters are described first.

### AliasId (p. 193)

Unique identifier for an alias associated with the fleet to search for active game sessions. Each request must reference either a fleet ID or alias ID, but not both.

- **Type:** String
- **Pattern:** ^alias-\S+
- **Required:** No

### FilterExpression (p. 193)

String containing the search criteria for the session search. If no filter expression is included, the request returns results for all game sessions in the fleet that are in **ACTIVE** status.

A filter expression can contain one or multiple conditions. Each condition consists of the following:

- **Operand** -- Name of a game session attribute. Valid values are gameSessionName, gameSessionId, gameSessionProperties, maximumSessions, creationTimeMillis, playerSessionCount, hasAvailablePlayerSessions.
- **Comparator** -- Valid comparators are: =, <>, <, <=, >=.
- **Value** -- Value to be searched for. Values may be numbers, boolean values (true/false) or strings depending on the operand. String values are case sensitive and must be enclosed in single quotes. Special characters must be escaped. Boolean and string values can only be used with the comparators = and <>. For example, the following filter expression searches on gameSessionName: "FilterExpression": "gameSessionName = 'Matt\'s Awesome Game 1'".

To chain multiple conditions in a single expression, use the logical keywords **AND**, **OR**, and **NOT** and parentheses as needed. For example: `x AND y AND NOT z, NOT (x OR y)`.

Session search evaluates conditions from left to right using the following precedence rules:

1. =, <>, <, >, <=, >=

API Version 2015-10-01

193
2. Parentheses
3. NOT
4. AND
5. OR

For example, this filter expression retrieves game sessions hosting at least ten players that have an
open player slot: "maximumSessions>=10 AND hasAvailablePlayerSessions=true".

Type: String
Required: No

FleetId (p. 193)
Unique identifier for a fleet to search for active game sessions. Each request must reference either a
fleet ID or alias ID, but not both.

Type: String
Pattern: ^fleet-\S+
Required: No

Limit (p. 193)
Maximum number of results to return. Use this parameter with NextToken to get results as a set of
sequential pages. The maximum number of results returned is 20, even if this value is not set or is
set higher than 20.

Type: Integer
Valid Range: Minimum value of 1.
Required: No

NextToken (p. 193)
Token that indicates the start of the next sequential page of results. Use the token that is returned
with a previous call to this action. To start at the beginning of the result set, do not specify a value.

Type: String
Required: No

SortExpression (p. 193)
Instructions on how to sort the search results. If no sort expression is included, the request returns
results in random order. A sort expression consists of the following elements:

- **Operand** -- Name of a game session attribute. Valid values are gameSessionName,
gameSessionId, gameSessionProperties, maximumSessions, creationTimeMillis,
playerSessionCount, hasAvailablePlayerSessions.
- **Order** -- Valid sort orders are ASC (ascending) and DESC (descending).

For example, this sort expression returns the oldest active sessions first: "SortExpression":
"creationTimeMillis ASC". Results with a null value for the sort operand are returned at the
end of the list.

Type: String

Required: No

Response Syntax

```json
{
  "GameSessions": [
    {
      "CreationTime": number,
      "CreatorId": "string",
      "CurrentPlayerSessionCount": number,
      "FleetId": "string",
      "GameProperties": [
        {
          "Key": "string",
          "Value": "string"
        }
      ],
      "GameSessionData": "string",
      "GameSessionId": "string",
      "IpAddress": "string",
      "MatchmakerData": "string",
      "MaximumPlayerSessionCount": number,
      "Name": "string",
      "PlayerSessionCreationPolicy": "string",
      "Port": number,
      "Status": "string",
      "StatusReason": "string",
      "TerminationTime": number
    }
  ],
  "NextToken": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**GameSessions (p. 195)**

Collection of objects containing game session properties for each session matching the request.

Type: Array of GameSession (p. 290) objects

**NextToken (p. 195)**

Token that indicates where to resume retrieving results on the next call to this action. If no token is returned, these results represent the end of the list.

Type: String


Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).
InternalServiceException

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

InvalidRequestException

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

NotFoundException

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

TerminalRoutingStrategyException

The service is unable to resolve the routing for a particular alias because it has a terminal RoutingStrategy (p. 329) associated with it. The message returned in this exception is the message defined in the routing strategy itself. Such requests should only be retried if the routing strategy for the specified alias is modified.

HTTP Status Code: 400

UnauthorizedException

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

Examples

Search game sessions

This example finds all game sessions with at least two players already connected. We recommend that all game session searches also filter out active game sessions that are not accepting new players.

HTTP requests are authenticated using an AWS Signature Version 4 signature in the Authorization header field.

Sample Request

```
POST / HTTP/1.1
Host: gamelift.us-west-2.amazonaws.com;
Accept-Encoding: identity
Content-Length: 135
User-Agent: aws-cli/1.11.36 Python/2.7.9 Windows/7 botocore/1.4.93
Content-Type: application/x-amz-json-1.0
Authorization: AWS4-HMAC-SHA256 Credential=AKIAIOSFODNN7EXAMPLE/20170406/us-west-2/
gamelift/aws4_request, SignedHeaders=content-type;host;x-amz-date;x-amz-target,
Signature=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
X-Amz-Date: 20170406T004805Z
X-Amz-Target: GameLift.SearchGameSessions

{"AliasId": "MAG-base",
 "FilterExpression": "playerSessionCount>=2 AND hasAvailablePlayerSessions=true",
 "Limit": 2
```
Search and sort game sessions

This example finds all game sessions that allow 20 or more players and are currently accepting new players. Results are sorted so that the newest game sessions are returned first.

HTTP requests are authenticated using an AWS Signature Version 4 signature in the Authorization header field.
Sample Request

POST / HTTP/1.1
Host: gamelift.us-west-2.amazonaws.com;
Accept-Encoding: identity
Content-Length: 198
User-Agent: aws-cli/1.11.36 Python/2.7.9 Windows/7 botocore/1.4.93
Content-Type: application/x-amz-json-1.0
Authorization: AWS4-HMAC-SHA256  Credential=AKIAIOSFODNN7EXAMPLE/20170406/us-west-2/
gamelift/aws4_request, SignedHeaders=content-type;host;x-amz-date;x-amz-target,
Signature=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
X-Amz-Date: 20170406T004805Z
X-Amz-Target: GameLift.SearchGameSessions

{"FleetId": "fleet-9999ffff-88ee-77dd-66cc-5555bbbb44aa",
 "FilterExpression": "maximumSessions>=20 AND hasAvailablePlayerSessions=true",
 "SortExpression": "creationTimeMillis DESC"
 "Limit": 2
}

CLI syntax:
aws gamelift search-game-sessions --fleet-id
"fleet-9999ffff-88ee-77dd-66cc-5555bbbb44aa" --filter-expression "maximumSessions=20 AND
hasAvailablePlayerSessions=true" --sort-expression "creationTimeMillis DESC"

Sample Response

HTTP/1.1 200 OK
x-amzn-RequestId: b34f8665-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 1184
Date: Thu, 06 Apr 2017 00:48:07 GMT

{
 "GameSessions": [
 {"CreationTime": 1469498497.792,
  "CurrentPlayerSessionCount": 3,
  "FleetId": "fleet-9999ffff-88ee-77dd-66cc-5555bbbb44aa",
  "GameProperties": [
   {"Key": "difficulty",
    "Value": "Hard"},
   {"Key": "gameMap",
    "Value": "Dystopia"},
   {"Key": "gameMode",
    "Value": "Brawl"}
  ],
  "GameSessionId": "gsess-7777dddd-55ee-66ff-5555bbbb44aa",
  "IpAddress": "192.0.2.0",
  "MaximumPlayerSessionCount": 20,
  "Name": "Matt's Awesome Game win456",
  "Port": "8080",
  "Status": "ACTIVE"
 },
 {"CreationTime": 1469498468.057,
  "CurrentPlayerSessionCount": ,
  "FleetId": "fleet-9999ffff-88ee-77dd-66cc-5555bbbb44aa",
  "GameProperties": [
   {"Key": "difficulty",
    "Value": "Easy"},
   {"Key": "gameMap",
    "Value": "Snowfall"},
   {"Key": "gameMode",
    "Value": "Brawl"}
  ]
}
Search game sessions by custom game data

This example searches for game sessions based on game map and game mode information, which is stored as custom data in the GameSession object. The game session property GameProperty contains custom data formatted as key:value pairs. In this example, we want to find all game sessions where the game mode is free-for-all (key:value pair "gameMode:Ffa") and set in either of two game maps (key:value pair "gameMap:Suzuka" or "gameMap:Silverstone"). Results are sorted by difficulty levels (key "difficulty", values "Easy", Normal" "Hard" "Insane"). Note: custom data is evaluated as a string, so the sorted results will be listed by the alphabetic order of the difficulty values.

HTTP requests are authenticated using an AWS Signature Version 4 signature in the Authorization header field.

Sample Request

POST / HTTP/1.1
Host: gamelift.us-west-2.amazonaws.com;
Accept-Encoding: identity
Content-Length: 198
User-Agent: aws-cli/1.11.36 Python/2.7.9 Windows/7 botocore/1.4.93
Content-Type: application/x-amz-json-1.0
Authorization: AWS4-HMAC-SHA256  Credential=AKIAIOSFODNN7EXAMPLE/20170406/us-west-2/gamelift/aws4_request, SignedHeaders=content-type;host;x-amz-date;x-amz-target, Signature=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
X-Amz-Date: 20170406T004805Z
X-Amz-Target: GameLift.SearchGameSessions

    "FleetId": "fleet-9999ffff-88ee-77dd-66cc-5555bbbb44aa",
    "FilterExpression": "gameSessionProperties.gameMode = 'Ffa' AND gameSessionProperties.gameMap = 'Suzuka' OR gameSessionProperties.gameMap = 'Silverstone'",
    "SortExpression": "gameSessionProperties.difficulty ASC"
    "Limit": 2

CLI syntax:
    aws gamelift search-game-sessions --fleet-id "9999ffff-88ee-77dd-66cc-5555bbbb44aa" --filter-expression "gameSessionProperties.gameMode = 'Ffa' AND gameSessionProperties.gameMap = 'Suzuka' OR gameSessionProperties.gameMap = 'Silverstone'" --sort-expression "gameSessionProperties.difficulty DESC"

Sample Response

HTTP/1.1 200 OK
x-amzn-RequestId: b34f8665-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 1184
Date: Thu, 06 Apr 2017 00:48:07 GMT
{  "GameSessions": [    {"CreationTime": 1469498468.057,    "CurrentPlayerSessionCount": 5,    "FleetId": "fleet-9999ffff-88ee-77dd-66cc-5555bbbb44aa",    "GameProperties": [      {"Key": "difficulty",      "Value": "Easy"},      {"Key": "gameMap",      "Value": "Suzuka"},      {"Key": "gameMode",      "Value": "Ffa"    ],    "GameSessionId": "gsess-4444dddd-55ee-66ff-77aa-8888bbbb99cc",    "IpAddress": "192.0.2.0",    "MaximumPlayerSessionCount": 10,    "Name": "Matt's Awesome Game win123",    "Port": "8080",    "Status": "ACTIVE"    },    {"CreationTime": 1469498497.792,    "CurrentPlayerSessionCount": 3,    "FleetId": "fleet-9999ffff-88ee-77dd-66cc-5555bbbb44aa",    "GameProperties": [      {"Key": "difficulty",      "Value": "Normal"},      {"Key": "gameMap",      "Value": "Silverstone"},      {"Key": "gameMode",      "Value": "Ffa"    ],    "GameSessionId": "gsess-7777dddd-55ee-66ff-44aa-8888bbbb99cc",    "IpAddress": "192.0.2.0",    "MaximumPlayerSessionCount": 10,    "Name": "Matt's Awesome Game win456",    "Port": "8080",    "Status": "ACTIVE"    }  ]}

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
StartFleetActions

Resumes activity on a fleet that was suspended with StopFleetActions (p. 219). Currently, this operation is used to restart a fleet's auto-scaling activity.

To start fleet actions, specify the fleet ID and the type of actions to restart. When auto-scaling fleet actions are restarted, Amazon GameLift once again initiates scaling events as triggered by the fleet's scaling policies. If actions on the fleet were never stopped, this operation will have no effect. You can view a fleet's stopped actions using DescribeFleetAttributes (p. 93).

Operations related to fleet capacity scaling include:

- DescribeFleetCapacity (p. 97)
- UpdateFleetCapacity (p. 238)
- DescribeEC2InstanceLimits (p. 90)
- Manage scaling policies:
  - PutScalingPolicy (p. 178) (auto-scaling)
  - DescribeScalingPolicies (p. 147) (auto-scaling)
  - DeleteScalingPolicy (p. 77) (auto-scaling)
- Manage fleet actions:
  - StartFleetActions (p. 201)
  - StopFleetActions (p. 219)

Request Syntax

```
{
  "Actions": [ "string" ],
  "FleetId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**
In the following list, the required parameters are described first.

**Actions (p. 201)**

List of actions to restart on the fleet.

Type: Array of strings

Array Members: Fixed number of 1 item.

Valid Values: AUTO_SCALING

Required: Yes

**FleetId (p. 201)**

Unique identifier for a fleet
Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

InternalServiceException

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

InvalidRequestException

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

NotFoundException

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

UnauthorizedException

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

Example

Restart fleet auto-scaling activity

In this example, we want resume use of all scaling policies that are defined for a specified fleet. Once started, the scaling policies will immediately begin tracking their respective metrics.

HTTP requests are authenticated using an AWS Signature Version 4 signature in the Authorization header field.

Sample Request

```plaintext
POST / HTTP/1.1
Host: gamelift.us-west-2.amazonaws.com;
Accept-Encoding: identity
Content-Length: 208
User-Agent: aws-cli/1.11.36 Python/2.7.9 Windows/7 botocore/1.4.93
```
Content-Type: application/x-amz-json-1.0
Authorization: AWS4-HMAC-SHA256  Credential=AKIAIOSFODNN7EXAMPLE/20170406/us-west-2/gamelift/aws4_request, SignedHeaders=content-type;host;x-amz-date;x-amz-target, Signature=wJalrXUtnFEM1/K7MDENG/bPxRfiCYEXAMPLEKEY
X-Amz-Date: 20170406T004805Z
X-Amz-Target: GameLift.StartFleetActions

{
    "FleetId": "fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
    "Actions": ["AUTO_SCALING"]
}

Sample Response

HTTP/1.1 200 OK undefined

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
StartGameSessionPlacement

Places a request for a new game session in a queue (see CreateGameSessionQueue (p. 37)). When processing a placement request, Amazon GameLift searches for available resources on the queue's destinations, scanning each until it finds resources or the placement request times out.

A game session placement request can also request player sessions. When a new game session is successfully created, Amazon GameLift creates a player session for each player included in the request.

When placing a game session, by default Amazon GameLift tries each fleet in the order they are listed in the queue configuration. Ideally, a queue's destinations are listed in preference order.

Alternatively, when requesting a game session with players, you can also provide latency data for each player in relevant regions. Latency data indicates the performance lag a player experiences when connected to a fleet in the region. Amazon GameLift uses latency data to reorder the list of destinations to place the game session in a region with minimal lag. If latency data is provided for multiple players, Amazon GameLift calculates each region's average lag for all players and reorders to get the best game play across all players.

To place a new game session request, specify the following:

- The queue name and a set of game session properties and settings
- A unique ID (such as a UUID) for the placement. You use this ID to track the status of the placement request
- (Optional) A set of IDs and player data for each player you want to join to the new game session
- Latency data for all players (if you want to optimize game play for the players)

If successful, a new game session placement is created.

To track the status of a placement request, call DescribeGameSessionPlacement (p. 117) and check the request's status. If the status is FULFILLED, a new game session has been created and a game session ARN and region are referenced. If the placement request times out, you can resubmit the request or retry it with a different queue.

Game-session-related operations include:

- CreateGameSession (p. 31)
- DescribeGameSessions (p. 123)
- DescribeGameSessionDetails (p. 113)
- SearchGameSessions (p. 192)
- UpdateGameSession (p. 246)
- GetGameSessionLogUrl (p. 157)
- Game session placements
  - StartGameSessionPlacement (p. 204)
  - DescribeGameSessionPlacement (p. 117)
  - StopGameSessionPlacement (p. 222)

Request Syntax

```json
{
  "DesiredPlayerSessions": [
    {
      "PlayerData": "string",
```

API Version 2015-10-01

204
Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**
In the following list, the required parameters are described first.

**GameSessionQueueName (p. 204)**
Name of the queue to use to place the new game session.

**Type:** String

**Length Constraints:** Minimum length of 1. Maximum length of 128.

**Pattern:** [a-zA-Z0-9-]+

**Required:** Yes

**MaximumPlayerSessionCount (p. 204)**
Maximum number of players that can be connected simultaneously to the game session.

**Type:** Integer

**Valid Range:** Minimum value of 0.

**Required:** Yes

**PlacementId (p. 204)**
Unique identifier to assign to the new game session placement. This value is developer-defined. The value must be unique across all regions and cannot be reused unless you are resubmitting a canceled or timed-out placement request.

**Type:** String

**Length Constraints:** Minimum length of 1. Maximum length of 48.
DesiredPlayerSessions (p. 204)

Set of information on each player to create a player session for.

Type: Array of DesiredPlayerSession (p. 273) objects

Required: No

GameProperties (p. 204)

Set of custom properties for a game session, formatted as key:value pairs. These properties are passed to a game server process in the GameSession (p. 290) object with a request to start a new game session (see Start a Game Session).

Type: Array of GameProperty (p. 289) objects

Array Members: Maximum number of 16 items.

Required: No

GameSessionData (p. 204)

Set of custom game session properties, formatted as a single string value. This data is passed to a game server process in the GameSession (p. 290) object with a request to start a new game session (see Start a Game Session).

Type: String


Required: No

GameSessionName (p. 204)

Descriptive label that is associated with a game session. Session names do not need to be unique.

Type: String


Required: No

PlayerLatencies (p. 204)

Set of values, expressed in milliseconds, indicating the amount of latency that a player experiences when connected to AWS regions. This information is used to try to place the new game session where it can offer the best possible gameplay experience for the players.

Type: Array of PlayerLatency (p. 323) objects

Required: No

Response Syntax

```
{
    "GameSessionPlacement": {
        "EndTime": number,
        "GameProperties": [
            {
            ...
```

API Version 2015-10-01
206
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

GameSessionPlacement (p. 206)

Object that describes the newly created game session placement. This object includes all the information provided in the request, as well as start/end time stamps and placement status.

Type: GameSessionPlacement (p. 297) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

InternalServiceException

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

InvalidRequestException

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.
Example

Request a new game session placement with player latency data

This example starts a new game session placement. The request calls for player sessions for two players, and provides each player's latency data for two regions.

Amazon GameLift uses the latency data provided to determine what order to use when looking for a fleet to host the new game session. It does this by calculating the average player latency for each region and ordering the queue's destinations starting with the lowest average latency. If the queue "matchmaker-queue" has a latency policy, however, things may change. For example, let's say matchmaker-queue has a policy that caps latency at 130 milliseconds for 60 seconds, followed by no cap. In this scenario, using the sample request below, the following sequence plays out:

1. Amazon GameLift calculates average latency for each region: \( \text{us-east-1} = 110 \) and \( \text{us-west-2} = 100 \).
2. Amazon GameLift reorders the queue's destinations based on lowest average latency, and prioritizes destinations in region \( \text{us-west-2} \).
3. The queue has a latency cap of 130 ms in force for the first 60 seconds of a placement. Amazon GameLift looks for any individual latency values that are greater than 130 ms. There is one: Player 2 reports a 150 ms latency when connected to region \( \text{us-west-2} \). As a result, Amazon GameLift temporarily drops all \( \text{us-west-2} \) fleets as valid destinations.
4. Amazon GameLift tries to place the new game session on fleets in region \( \text{us-east-1} \), followed by fleets in regions with no latency information (if any). If available resources are found, the game session is placed and the request fulfilled.
5. If no available resources are found, Amazon GameLift starts a new round of placement attempts, restarting at step 3. If 60 seconds have passed and the latency policy is no longer in force, then fleets in region \( \text{us-west-2} \) are once more valid destinations -- and are preferred based on their low average latency.
6. Amazon GameLift continues to attempt to place the new game session until it is successful or until the queue's timeout limit is reached.

HTTP requests are authenticated using an AWS Signature Version 4 signature in the Authorization header field.

Sample Request

```
POST / HTTP/1.1
Host: gamelift.us-west-2.amazonaws.com;
Accept-Encoding: identity
Content-Length: 824
```

API Version 2015-10-01
User-Agent: aws-cli/1.11.36 Python/2.7.9 Windows/7 botocore/1.4.93
Content-Type: application/x-amz-json-1.0
Authorization: AWS4-HMAC-SHA256  Credential=AKIAIOSFODNN7EXAMPLE/20170406/us-west-2/gameli/aw4_request, SignedHeaders=content-type;host;x-amz-date;x-amz-target, Signature=wJalrXUtFEMI/K7MDENG/bPxFriCYEXAMPLEKEY
X-Amz-Date: 20170406T004805Z
X-Amz-Target: GameLift.StartGameSessionPlacement

{
  "DesiredPlayerSessions": [
    { "PlayerData": "level:10", "PlayerId": "player1" },
    { "PlayerData": "level:11", "PlayerId": "player2" },
  ],
  "GameProperties": [
    { "Key": "map", "Value": "winter" }
  ],
  "GameSessionName": "matchmaker-1234567890",
  "GameSessionQueueName": "matchmaker-queue",
  "MaximumPlayerSessionCount": 4,
  "PlacementId": "Place-12345",
  "PlayerLatencies": [
    { "LatencyInMilliseconds": 100, "PlayerId": "player1", "RegionIdentifier": "us-east-1" },
    { "LatencyInMilliseconds": 50, "PlayerId": "player1", "RegionIdentifier": "us-west-2" },
    { "LatencyInMilliseconds": 120, "PlayerId": "player2", "RegionIdentifier": "us-east-1" },
    { "LatencyInMilliseconds": 150, "PlayerId": "player2", "RegionIdentifier": "us-west-2" }
  ]
}

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
StartMatchBackfill

Finds new players to fill open slots in an existing game session. This operation can be used to add players to matched games that start with fewer than the maximum number of players or to replace players when they drop out. By backfilling with the same matchmaker used to create the original match, you ensure that new players meet the match criteria and maintain a consistent experience throughout the game session. You can backfill a match anytime after a game session has been created.

To request a match backfill, specify a unique ticket ID, the existing game session's ARN, a matchmaking configuration, and a set of data that describes all current players in the game session. If successful, a match backfill ticket is created and returned with status set to QUEUED. The ticket is placed in the matchmaker's ticket pool and processed. Track the status of the ticket to respond as needed. For more detail how to set up backfilling, see Backfill Existing Games with FlexMatch.

The process of finding backfill matches is essentially identical to the initial matchmaking process. The matchmaker searches the pool and groups tickets together to form potential matches, allowing only one backfill ticket per potential match. Once a match is formed, the matchmaker creates player sessions for the new players. All tickets in the match are updated with the game session's connection information, and the GameSession (p. 290) object is updated to include matchmaker data on the new players. For more detail on how match backfill requests are processed, see How Amazon GameLift FlexMatch Works.

Matchmaking-related operations include:

- StartMatchmaking (p. 214)
- DescribeMatchmaking (p. 131)
- StopMatchmaking (p. 225)
- AcceptMatch (p. 7)
- StartMatchBackfill (p. 210)

Request Syntax

```
{
   "ConfigurationName": "string",
   "GameSessionArn": "string",
   "Players": [
   {
      "LatencyInMs": {
         "string" : number
      },
      "PlayerAttributes": {
         "string" : {
            "N": number,
            "S": "string",
            "SDM": {
               "string" : number
            },
            "SL": [ "string" ]
         }
      },
      "PlayerId": "string",
      "Team": "string"
   }
   ],
   "TicketId": "string"
}
```
Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**
In the following list, the required parameters are described first.

**ConfigurationName (p. 210)**

Name of the matchmaker to use for this request. The name of the matchmaker that was used with the original game session is listed in the GameSession (p. 290) object, MatchmakerData property. This property contains a matchmaking configuration ARN value, which includes the matchmaker name. (In the ARN value "arn:aws:gamelift:us-west-2:111122223333:matchmakingconfiguration/MM-4v4", the matchmaking configuration name is "MM-4v4"). Use only the name for this parameter.

Type: String


Pattern: [a-zA-Z0-9-\./]+

Required: Yes

**GameSessionArn (p. 210)**

Amazon Resource Name (ARN) that is assigned to a game session and uniquely identifies it.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: [a-zA-Z0-9:/-]+

Required: Yes

**Players (p. 210)**

Match information on all players that are currently assigned to the game session. This information is used by the matchmaker to find new players and add them to the existing game.

- PlayerID, PlayerAttributes, Team - This information is maintained in the GameSession (p. 290) object, MatchmakerData property, for all players who are currently assigned to the game session. The matchmaker data is in JSON syntax, formatted as a string. For more details, see Match Data.
- LatencyInMs - If the matchmaker uses player latency, include a latency value, in milliseconds, for the region that the game session is currently in. Do not include latency values for any other region.

Type: Array of Player (p. 321) objects

Required: Yes

**TicketId (p. 210)**

Unique identifier for a matchmaking ticket. If no ticket ID is specified here, Amazon GameLift will generate one in the form of a UUID. Use this identifier to track the match backfill ticket status and retrieve match results.

Type: String

Response Syntax

```json
{
   "MatchmakingTicket": {
      "ConfigurationName": "string",
      "EndTime": number,
      "EstimatedWaitTime": number,
      "GameSessionConnectionInfo": {
         "GameSessionArn": "string",
         "IpAddress": "string",
         "MatchedPlayerSessions": [
            {
               "PlayerId": "string",
               "PlayerSessionId": "string"
            }
         ],
         "Port": number
      },
      "Players": [
         {
            "LatencyInMs": {
               "string" : number
            },
            "PlayerAttributes": {
               "string" : {
                  "N": number,
                  "S": "string",
                  "SDM": {
                     "string" : number
                  },
                  "SL": [ "string" ]
               }
            },
            "PlayerId": "string",
            "Team": "string"
         }
      ],
      "StartTime": number,
      "Status": "string",
      "StatusMessage": "string",
      "StatusReason": "string",
      "TicketId": "string"
   }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**MatchmakingTicket (p. 212)**

Ticket representing the backfill matchmaking request. This object includes the information in the request, ticket status, and match results as generated during the matchmaking process.

Type: MatchmakingTicket (p. 317) object
Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

InternalServiceException

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

InvalidRequestException

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

NotFoundException

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

UnsupportedRegionException

The requested operation is not supported in the region specified.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
StartMatchmaking

Uses FlexMatch to create a game match for a group of players based on custom matchmaking rules, and starts a new game for the matched players. Each matchmaking request specifies the type of match to build (team configuration, rules for an acceptable match, etc.). The request also specifies the players to find a match for and where to host the new game session for optimal performance. A matchmaking request might start with a single player or a group of players who want to play together. FlexMatch finds additional players as needed to fill the match. Match type, rules, and the queue used to place a new game session are defined in a MatchmakingConfiguration. For complete information on setting up and using FlexMatch, see the topic Adding FlexMatch to Your Game.

To start matchmaking, provide a unique ticket ID, specify a matchmaking configuration, and include the players to be matched. You must also include a set of player attributes relevant for the matchmaking configuration. If successful, a matchmaking ticket is returned with status set to QUEUED. Track the status of the ticket to respond as needed and acquire game session connection information for successfully completed matches.

Tracking ticket status -- A couple of options are available for tracking the status of matchmaking requests:

- Polling -- Call DescribeMatchmaking. This operation returns the full ticket object, including current status and (for completed tickets) game session connection info. We recommend polling no more than once every 10 seconds.
- Notifications -- Get event notifications for changes in ticket status using Amazon Simple Notification Service (SNS). Notifications are easy to set up (see CreateMatchmakingConfiguration (p. 41)) and typically deliver match status changes faster and more efficiently than polling. We recommend that you use polling to back up to notifications (since delivery is not guaranteed) and call DescribeMatchmaking only when notifications are not received within 30 seconds.

Processing a matchmaking request -- FlexMatch handles a matchmaking request as follows:

1. Your client code submits a StartMatchmaking request for one or more players and tracks the status of the request ticket.
2. FlexMatch uses this ticket and others in process to build an acceptable match. When a potential match is identified, all tickets in the proposed match are advanced to the next status.
3. If the match requires player acceptance (set in the matchmaking configuration), the tickets move into status REQUIRES_ACCEPTANCE. This status triggers your client code to solicit acceptance from all players in every ticket involved in the match, and then call AcceptMatch (p. 7) for each player. If any player rejects or fails to accept the match before a specified timeout, the proposed match is dropped (see AcceptMatch for more details).
4. Once a match is proposed and accepted, the matchmaking tickets move into status PLACING. FlexMatch locates resources for a new game session using the game session queue (set in the matchmaking configuration) and creates the game session based on the match data.
5. When the match is successfully placed, the matchmaking tickets move into COMPLETED status. Connection information (including game session endpoint and player session) is added to the matchmaking tickets. Matched players can use the connection information to join the game.

Matchmaking-related operations include:

- StartMatchmaking (p. 214)
- DescribeMatchmaking (p. 131)
- StopMatchmaking (p. 225)
- AcceptMatch (p. 7)
Request Syntax

```
{  
  "ConfigurationName": "string",
  "Players": [
    {
      "LatencyInMs": {
        "string": number
      },
      "PlayerAttributes": {
        "string": {
          "M": number,
          "S": "string",
          "SDM": {
            "string": number
          },
          "SL": [ "string" ]
        }
      },
      "PlayerId": "string",
      "Team": "string"
    }
  ],
  "TicketId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**  
In the following list, the required parameters are described first.

**ConfigurationName (p. 215)**

Name of the matchmaking configuration to use for this request. Matchmaking configurations must exist in the same region as this request.

Type: String


Pattern: [a-zA-Z0-9-\.]+

Required: Yes

**Players (p. 215)**

Information on each player to be matched. This information must include a player ID, and may contain player attributes and latency data to be used in the matchmaking process. After a successful match, Player objects contain the name of the team the player is assigned to.

Type: Array of Player (p. 321) objects
Required: Yes

**TicketId (p. 215)**

Unique identifier for a matchmaking ticket. If no ticket ID is specified here, Amazon GameLift will generate one in the form of a UUID. Use this identifier to track the matchmaking ticket status and retrieve match results.

Type: String


Pattern: [a-zA-Z0-9-\.]+

Required: No

**Response Syntax**

```json
{
    "MatchmakingTicket": {
        "ConfigurationName": "string",
        "EndTime": number,
        "EstimatedWaitTime": number,
        "GameSessionConnectionInfo": {
            "GameSessionArn": "string",
            "IpAddress": "string",
            "MatchedPlayerSessions": [
                {
                    "PlayerId": "string",
                    "PlayerSessionId": "string"
                }
            ],
            "Port": number
        },
        "Players": [
            {
                "LatencyInMs": {
                    "string": number
                },
                "PlayerAttributes": {
                    "string": {
                        "N": number,
                        "S": "string",
                        "SDM": {
                            "string": number
                        },
                        "SL": [ "string" ]
                    }
                },
                "PlayerId": "string",
                "Team": "string"
            }
        ],
        "StartTime": number,
        "Status": "string",
        "StatusMessage": "string",
        "StatusReason": "string",
        "TicketId": "string"
    }
}
```
Response Elements

If the action is successful, the service sends back an HTTP 200 response. The following data is returned in JSON format by the service.

**MatchmakingTicket (p. 216)**

Ticket representing the matchmaking request. This object include the information included in the request, ticket status, and match results as generated during the matchmaking process.

Type: `MatchmakingTicket (p. 317)` object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

**InternalServiceException**

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

**InvalidRequestException**

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

**NotFoundException**

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

**UnsupportedRegionException**

The requested operation is not supported in the region specified.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
• AWS SDK for Ruby V2
StopFleetActions

Suspend activity on a fleet. Currently, this operation is used to stop a fleet's auto-scaling activity. It is used to temporarily stop scaling events triggered by the fleet's scaling policies. The policies can be retained and auto-scaling activity can be restarted using StartFleetActions (p. 201). You can view a fleet's stopped actions using DescribeFleetAttributes (p. 93).

To stop fleet actions, specify the fleet ID and the type of actions to suspend. When auto-scaling fleet actions are stopped, Amazon GameLift no longer initiates scaling events except to maintain the fleet's desired instances setting (FleetCapacity (p. 285). Changes to the fleet's capacity must be done manually using UpdateFleetCapacity (p. 238).

Request Syntax

```json
{
   "Actions": [ "string" ],
   "FleetId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

Note

In the following list, the required parameters are described first.

**Actions (p. 219)**

List of actions to suspend on the fleet.

Type: Array of strings

Array Members: Fixed number of 1 item.

Valid Values: AUTO_SCALING

Required: Yes

**FleetId (p. 219)**

Unique identifier for a fleet

Type: String

Pattern: ^fleet-\S+

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).
InternalServiceException

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

InvalidRequestException

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

NotFoundException

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

UnauthorizedException

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

Example

Stop fleet auto-scaling activity

In this example, we want to suspend all scaling policies that are defined for a specified fleet. Once the policies are suspended, fleet capacity will remain at the last "desired instances" setting unless we adjust capacity manually by calling UpdateFleetCapacity (p. 238).

HTTP requests are authenticated using an AWS Signature Version 4 signature in the Authorization header field.

Sample Request

```plaintext
POST / HTTP/1.1
Host: gamelift.us-west-2.amazonaws.com;
Accept-Encoding: identity
Content-Length: 208
User-Agent: aws-cli/1.11.36 Python/2.7.9 Windows/7 botocore/1.4.93
Content-Type: application/x-amz-json-1.0
Authorization: AWS4-HMAC-SHA256  Credential=AKIAIOSFODNN7EXAMPLE/20170406/us-west-2/
gamelift/aws4_request, SignedHeaders=content-type;host;x-amz-date;x-amz-target,
Signature=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
X-Amz-Date: 20170406T004805Z
X-Amz-Target: GameLift.StopFleetActions
{
  "FleetId": "fleet-2222bbbb-33cc-44dd-55ee-6666ffff77aa",
  "Actions": ["AUTO_SCALING"]
}
```

Sample Response

```
HTTP/1.1 200 OK undefined
```
See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
StopGameSessionPlacement

Cancels a game session placement that is in PENDING status. To stop a placement, provide the placement ID values. If successful, the placement is moved to CANCELLED status.

Game-session-related operations include:

- CreateGameSession (p. 31)
- DescribeGameSessions (p. 123)
- DescribeGameSessionDetails (p. 113)
- SearchGameSessions (p. 192)
- UpdateGameSession (p. 246)
- GetGameSessionLogUrl (p. 157)
- Game session placements
  - StartGameSessionPlacement (p. 204)
  - DescribeGameSessionPlacement (p. 117)
  - StopGameSessionPlacement (p. 222)

Request Syntax

```json
{
  "PlacementId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**
In the following list, the required parameters are described first.

**PlacementId (p. 222)**

Unique identifier for a game session placement to cancel.

Type: String


Pattern: [a-zA-Z0-9-]+

Required: Yes

Response Syntax

```json
{
  "GameSessionPlacement": {
    "EndTime": number,
    "GameProperties": [
      
    ]
  }
}
```
Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

GameSessionPlacement (p. 222)

Object that describes the canceled game session placement, with CANCELLED status and an end time stamp.

Type: GameSessionPlacement (p. 297) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

InternalServiceException

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

InvalidRequestException

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.
HTTP Status Code: 400

**NotFoundException**

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

**UnauthorizedException**

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
StopMatchmaking

Cancels a matchmaking ticket that is currently being processed. To stop the matchmaking operation, specify the ticket ID. If successful, work on the ticket is stopped, and the ticket status is changed to CANCELLED.

Matchmaking-related operations include:

- StartMatchmaking (p. 214)
- DescribeMatchmaking (p. 131)
- StopMatchmaking (p. 225)
- AcceptMatch (p. 7)
- StartMatchBackfill (p. 210)

Request Syntax

```json
{
  "TicketId": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.

**TicketId (p. 225)**

Unique identifier for a matchmaking ticket.

Type: String


Pattern: [a-zA-Z0-9-\.]+

Required: Yes

Response Elements

If the action is successful, the service sends back an HTTP 200 response with an empty HTTP body.

Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

**InternalServiceException**

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.
HTTP Status Code: 500

**InvalidRequestException**

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

**NotFoundException**

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

**UnsupportedRegionException**

The requested operation is not supported in the region specified.

HTTP Status Code: 400

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
UpdateAlias

Updates properties for an alias. To update properties, specify the alias ID to be updated and provide the information to be changed. To reassign an alias to another fleet, provide an updated routing strategy. If successful, the updated alias record is returned.

Alias-related operations include:

- CreateAlias (p. 10)
- ListAliases (p. 165)
- DescribeAlias (p. 84)
- UpdateAlias (p. 227)
- DeleteAlias (p. 65)
- ResolveAlias (p. 189)

Request Syntax

```json
{
    "AliasId": "string",
    "Description": "string",
    "Name": "string",
    "RoutingStrategy": {
        "FleetId": "string",
        "Message": "string",
        "Type": "string"
    }
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.

**AliasId (p. 227)**

Unique identifier for a fleet alias. Specify the alias you want to update.

Type: String

Pattern: ^alias-\S+

Required: Yes

**Description (p. 227)**

Human-readable description of an alias.

Type: String


Required: No
Name (p. 227)

Descriptive label that is associated with an alias. Alias names do not need to be unique.

Type: String


Pattern: .\S.*

Required: No

RoutingStrategy (p. 227)

Object that specifies the fleet and routing type to use for the alias.

Type: RoutingStrategy (p. 329) object

Required: No

Response Syntax

```json
{
    "Alias": {
        "AliasArn": "string",
        "AliasId": "string",
        "CreationTime": number,
        "Description": "string",
        "LastUpdatedTime": number,
        "Name": "string",
        "RoutingStrategy": {
            "FleetId": "string",
            "Message": "string",
            "Type": "string"
        }
    }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

Alias (p. 228)

Object that contains the updated alias configuration.

Type: Alias (p. 266) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

InternalServiceException

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.
HTTP Status Code: 500
InvalidRequestException

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400
NotFoundException

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400
UnauthorizedException

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
UpdateBuild

Updates metadata in a build record, including the build name and version. To update the metadata, specify the build ID to update and provide the new values. If successful, a build object containing the updated metadata is returned.

Build-related operations include:

- CreateBuild (p. 13)
- ListBuilds (p. 168)
- DescribeBuild (p. 87)
- UpdateBuild (p. 230)
- DeleteBuild (p. 67)

Request Syntax

```json
{
  "BuildId": "string",
  "Name": "string",
  "Version": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

*Note*

In the following list, the required parameters are described first.

**BuildId (p. 230)**

Unique identifier for a build to update.

Type: String

Pattern: ^build-\S+

Required: Yes

**Name (p. 230)**

Descriptive label that is associated with a build. Build names do not need to be unique.

Type: String


Required: No

**Version (p. 230)**

Version that is associated with this build. Version strings do not need to be unique.

Type: String
Response Syntax

{  
  "Build": {  
    "BuildId": "string",
    "CreationTime": number,
    "Name": "string",
    "OperatingSystem": "string",
    "SizeOnDisk": number,
    "Status": "string",
    "Version": "string"
  }
}

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**Build (p. 231)**

Object that contains the updated build record.

Type: Build (p. 271) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

**InternalServiceException**

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

**InvalidRequestException**

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

**NotFoundException**

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

**UnauthorizedException**

The client failed authentication. Clients should not retry such requests.
HTTP Status Code: 400

Example

Change a build record

This example changes a build record from the name "My_Game_Server_Build_One" and version number "12345.678". These are the only elements of a build record that can be changed. The returned build record verifies that the changes were made.

HTTP requests are authenticated using an AWS Signature Version 4 signature in the Authorization header field.

Sample Request

```plaintext
POST / HTTP/1.1
Host: gamelift.us-west-2.amazonaws.com;
Accept-Encoding: identity
Content-Length: 131
User-Agent: aws-cli/1.11.36 Python/2.7.9 Windows/7 botocore/1.4.93
Content-Type: application/x-amz-json-1.0
Authorization: AWS4-HMAC-SHA256  Credential=AKIAIOSFODDN7EXAMPLE/20170406/us-west-2/
gamelift/aws4_request, SignedHeaders=content-type;host;x-amz-date;x-amz-target,
Signature=wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY
X-Amz-Date: 20170406T004805Z
X-Amz-Target: GameLift.UpdateBuild
{
  "BuildId": "build-111aaaaa-22bb-33cc-44dd-5555eeee66ff",
  "Name": "My_Game_Server_Build_Foo",
  "Version": "12345.f00"
}
```

Sample Response

```plaintext
HTTP/1.1 200 OK
x-amzn-RequestId: b34f8665-EXAMPLE
Content-Type: application/x-amz-json-1.1
Content-Length: 310
Date: Thu, 06 Apr 2017 00:48:07 GMT
{
  "Build": {
    "BuildId": "build-111aaaaa-22bb-33cc-44dd-5555eeee66ff",
    "CreationTime": 1496708916.18,
    "Name": "My_Game_Server_Build_Foo",
    "OperatingSystem": "AMAZON_LINUX",
    "SizeOnDisk": 0,
    "Status": "INITIALIZED",
    "Version": "12345.f00"
  }
}
```

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
See Also

- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
UpdateFleetAttributes

Updates fleet properties, including name and description, for a fleet. To update metadata, specify the fleet ID and the property values that you want to change. If successful, the fleet ID for the updated fleet is returned.

Fleet-related operations include:

- CreateFleet (p. 19)
- ListFleets (p. 173)
- DeleteFleet (p. 70)
- Describe fleets:
  - DescribeFleetAttributes (p. 93)
  - DescribeFleetCapacity (p. 97)
  - DescribeFleetPortSettings (p. 107)
  - DescribeFleetUtilization (p. 110)
  - DescribeRuntimeConfiguration (p. 144)
  - DescribeEC2InstanceLimits (p. 90)
  - DescribeFleetEvents (p. 101)
- Update fleets:
  - UpdateFleetAttributes (p. 234)
  - UpdateFleetCapacity (p. 238)
  - UpdateFleetPortSettings (p. 242)
  - UpdateRuntimeConfiguration (p. 258)
- Manage fleet actions:
  - StartFleetActions (p. 201)
  - StopFleetActions (p. 219)

Request Syntax

```json
{
  "Description": "string",
  "FleetId": "string",
  "MetricGroups": [ "string" ],
  "Name": "string",
  "NewGameSessionProtectionPolicy": "string",
  "ResourceCreationLimitPolicy": {
    "NewGameSessionsPerCreator": number,
    "PolicyPeriodInMinutes": number
  }
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.
FleetId (p. 234)
Unique identifier for a fleet to update attribute metadata for.
Type: String
Pattern: ^fleet-\S+
Required: Yes

Description (p. 234)
Human-readable description of a fleet.
Type: String
Required: No

MetricGroups (p. 234)
Names of metric groups to include this fleet in. Amazon CloudWatch uses a fleet metric group to aggregate metrics from multiple fleets. Use an existing metric group name to add this fleet to the group. Or use a new name to create a new metric group. A fleet can only be included in one metric group at a time.
Type: Array of strings
Array Members: Maximum number of 1 item.
Length Constraints: Minimum length of 1. Maximum length of 255.
Required: No

Name (p. 234)
Descriptive label that is associated with a fleet. Fleet names do not need to be unique.
Type: String
Required: No

NewGameSessionProtectionPolicy (p. 234)
Game session protection policy to apply to all new instances created in this fleet. Instances that already exist are not affected. You can set protection for individual instances using UpdateGameSession (p. 246).
- NoProtection -- The game session can be terminated during a scale-down event.
- FullProtection -- If the game session is in an ACTIVE status, it cannot be terminated during a scale-down event.
Type: String
Valid Values: NoProtection | FullProtection
Required: No

ResourceCreationLimitPolicy (p. 234)
Policy that limits the number of game sessions an individual player can create over a span of time.
Type: ResourceCreationLimitPolicy (p. 328) object
Required: No

Response Syntax

```
{
  "FleetId": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**FleetId (p. 236)**

Unique identifier for a fleet that was updated.

- Type: String
- Pattern: ^fleet-\S+

Errors

For information about the errors that are common to all actions, see [Common Errors (p. 347)].

**ConflictException**

The requested operation would cause a conflict with the current state of a service resource associated with the request. Resolve the conflict before retrying this request.

- HTTP Status Code: 400

**InternalServiceException**

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

- HTTP Status Code: 500

**InvalidFleetStatusException**

The requested operation would cause a conflict with the current state of a resource associated with the request and/or the fleet. Resolve the conflict before retrying.

- HTTP Status Code: 400

**InvalidRequestException**

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

- HTTP Status Code: 400

**LimitExceededException**

The requested operation would cause the resource to exceed the allowed service limit. Resolve the issue before retrying.
HTTP Status Code: 400

**NotFoundException**

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

**UnauthorizedException**

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
UpdateFleetCapacity

Updates capacity settings for a fleet. Use this action to specify the number of EC2 instances (hosts) that you want this fleet to contain. Before calling this action, you may want to call DescribeEC2InstanceLimits (p. 90) to get the maximum capacity based on the fleet's EC2 instance type.

Specify minimum and maximum number of instances. Amazon GameLift will not change fleet capacity to values fall outside of this range. This is particularly important when using auto-scaling (see PutScalingPolicy (p. 178)) to allow capacity to adjust based on player demand while imposing limits on automatic adjustments.

To update fleet capacity, specify the fleet ID and the number of instances you want the fleet to host. If successful, Amazon GameLift starts or terminates instances so that the fleet's active instance count matches the desired instance count. You can view a fleet's current capacity information by calling DescribeFleetCapacity (p. 97). If the desired instance count is higher than the instance type's limit, the "Limit Exceeded" exception occurs.

Fleet-related operations include:

- CreateFleet (p. 19)
- ListFleets (p. 173)
- DeleteFleet (p. 70)
- Describe fleets:
  - DescribeFleetAttributes (p. 93)
  - DescribeFleetCapacity (p. 97)
  - DescribeFleetPortSettings (p. 107)
  - DescribeFleetUtilization (p. 110)
  - DescribeRuntimeConfiguration (p. 144)
  - DescribeEC2InstanceLimits (p. 90)
  - DescribeFleetEvents (p. 101)
- Update fleets:
  - UpdateFleetAttributes (p. 234)
  - UpdateFleetCapacity (p. 238)
  - UpdateFleetPortSettings (p. 242)
  - UpdateRuntimeConfiguration (p. 258)
- Manage fleet actions:
  - StartFleetActions (p. 201)
  - StopFleetActions (p. 219)

Request Syntax

```json
{
  "DesiredInstances": number,
  "FleetId": "string",
  "MaxSize": number,
  "MinSize": number
}
```
Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**
In the following list, the required parameters are described first.

**FleetId (p. 238)**
Unique identifier for a fleet to update capacity for.
Type: String
Pattern: ^fleet-\S+
Required: Yes

**DesiredInstances (p. 238)**
Number of EC2 instances you want this fleet to host.
Type: Integer
Valid Range: Minimum value of 0.
Required: No

**MaxSize (p. 238)**
Maximum value allowed for the fleet's instance count. Default if not set is 1.
Type: Integer
Valid Range: Minimum value of 0.
Required: No

**MinSize (p. 238)**
Minimum value allowed for the fleet's instance count. Default if not set is 0.
Type: Integer
Valid Range: Minimum value of 0.
Required: No

Response Syntax

```
{
  "FleetId": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.
The following data is returned in JSON format by the service.
FleetId (p. 239)

Unique identifier for a fleet that was updated.

Type: String

Pattern: ^fleet-\S+

Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

ConflictException

The requested operation would cause a conflict with the current state of a service resource associated with the request. Resolve the conflict before retrying this request.

HTTP Status Code: 400

InternalServiceException

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

InvalidFleetStatusException

The requested operation would cause a conflict with the current state of a resource associated with the request and/or the fleet. Resolve the conflict before retrying.

HTTP Status Code: 400

InvalidRequestException

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

LimitExceededException

The requested operation would cause the resource to exceed the allowed service limit. Resolve the issue before retrying.

HTTP Status Code: 400

NotFoundException

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

UnauthorizedException

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:
• AWS Command Line Interface
• AWS SDK for .NET
• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V2
UpdateFleetPortSettings

Updates port settings for a fleet. To update settings, specify the fleet ID to be updated and list the permissions you want to update. List the permissions you want to add in `InboundPermissionAuthorizations`, and permissions you want to remove in `InboundPermissionRevocations`. Permissions to be removed must match existing fleet permissions. If successful, the fleet ID for the updated fleet is returned.

Fleet-related operations include:

- CreateFleet (p. 19)
- ListFleets (p. 173)
- DeleteFleet (p. 70)
- Describe fleets:
  - DescribeFleetAttributes (p. 93)
  - DescribeFleetCapacity (p. 97)
  - DescribeFleetPortSettings (p. 107)
  - DescribeFleetUtilization (p. 110)
  - DescribeRuntimeConfiguration (p. 144)
  - DescribeEC2InstanceLimits (p. 90)
  - DescribeFleetEvents (p. 101)
- Update fleets:
  - UpdateFleetAttributes (p. 234)
  - UpdateFleetCapacity (p. 238)
  - UpdateFleetPortSettings (p. 242)
  - UpdateRuntimeConfiguration (p. 258)
- Manage fleet actions:
  - StartFleetActions (p. 201)
  - StopFleetActions (p. 219)

Request Syntax

```json
{
  "FleetId": "string",
  "InboundPermissionAuthorizations": [
    {
      "FromPort": number,
      "IpRange": "string",
      "Protocol": "string",
      "ToPort": number
    }
  ],
  "InboundPermissionRevocations": [
    {
      "FromPort": number,
      "IpRange": "string",
      "Protocol": "string",
      "ToPort": number
    }
  ]
}
```

API Version 2015-10-01

242
Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**
In the following list, the required parameters are described first.

**FleetId (p. 242)**
Unique identifier for a fleet to update port settings for.
Type: String
Pattern: ^fleet-\S+
Required: Yes

**InboundPermissionAuthorizations (p. 242)**
Collection of port settings to be added to the fleet record.
Type: Array of IpPermission (p. 309) objects
Array Members: Maximum number of 50 items.
Required: No

**InboundPermissionRevocations (p. 242)**
Collection of port settings to be removed from the fleet record.
Type: Array of IpPermission (p. 309) objects
Array Members: Maximum number of 50 items.
Required: No

Response Syntax

```json
{
   "FleetId": "string"
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**FleetId (p. 243)**
Unique identifier for a fleet that was updated.
Type: String
Pattern: ^fleet-\S+
Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

**ConflictException**

The requested operation would cause a conflict with the current state of a service resource associated with the request. Resolve the conflict before retrying this request.

HTTP Status Code: 400

**InternalServiceException**

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

**InvalidFleetStatusException**

The requested operation would cause a conflict with the current state of a resource associated with the request and/or the fleet. Resolve the conflict before retrying.

HTTP Status Code: 400

**InvalidRequestException**

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

**LimitExceededException**

The requested operation would cause the resource to exceed the allowed service limit. Resolve the issue before retrying.

HTTP Status Code: 400

**NotFoundException**

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

**UnauthorizedException**

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
• AWS SDK for JavaScript
• AWS SDK for PHP V3
• AWS SDK for Python
• AWS SDK for Ruby V2
UpdateGameSession

Updates game session properties. This includes the session name, maximum player count, protection policy, which controls whether or not an active game session can be terminated during a scale-down event, and the player session creation policy, which controls whether or not new players can join the session. To update a game session, specify the game session ID and the values you want to change. If successful, an updated GameSession (p. 290) object is returned.

Game-session-related operations include:

- CreateGameSession (p. 31)
- DescribeGameSessions (p. 123)
- DescribeGameSessionDetails (p. 113)
- SearchGameSessions (p. 192)
- UpdateGameSession (p. 246)
- GetGameSessionLogUrl (p. 157)

Game session placements

- StartGameSessionPlacement (p. 204)
- DescribeGameSessionPlacement (p. 117)
- StopGameSessionPlacement (p. 222)

Request Syntax

```json
{
    "GameSessionId": "string",
    "MaximumPlayerSessionCount": number,
    "Name": "string",
    "PlayerSessionCreationPolicy": "string",
    "ProtectionPolicy": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.

**GameSessionId (p. 246)**

Unique identifier for the game session to update.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: [a-zA-Z0-9:\/-]+

Required: Yes
MaximumPlayerSessionCount (p. 246)

Maximum number of players that can be connected simultaneously to the game session.

Type: Integer

Valid Range: Minimum value of 0.

Required: No

Name (p. 246)

Descriptive label that is associated with a game session. Session names do not need to be unique.

Type: String


Required: No

PlayerSessionCreationPolicy (p. 246)

Policy determining whether or not the game session accepts new players.

Type: String

Valid Values: ACCEPT_ALL | DENY_ALL

Required: No

ProtectionPolicy (p. 246)

Game session protection policy to apply to this game session only.
- **NoProtection** -- The game session can be terminated during a scale-down event.
- **FullProtection** -- If the game session is in an ACTIVE status, it cannot be terminated during a scale-down event.

Type: String

Valid Values: NoProtection | FullProtection

Required: No

Response Syntax

```
{

  "GameSession": {
    "CreationTime": number,
    "CreatorId": "string",
    "CurrentPlayerSessionCount": number,
    "FleetId": "string",
    "GameProperties": [
      {
        "Key": "string",
        "Value": "string"
      }
    ],
    "GameSessionData": "string",
    "GameSessionId": "string",
    "IpAddress": "string",
    "MatchmakerData": "string",
    "MaximumPlayerSessionCount": number,
    "Name": "string"
  }
```

API Version 2015-10-01
247
"PlayerSessionCreationPolicy": "string",
"Port": number,
"Status": "string",
"StatusReason": "string",
"TerminationTime": number
}

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**GameSession (p. 247)**

Object that contains the updated game session metadata.

Type: GameSession (p. 290) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

**ConflictException**

The requested operation would cause a conflict with the current state of a service resource associated with the request. Resolve the conflict before retrying this request.

HTTP Status Code: 400

**InternalServiceException**

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

**InvalidGameSessionStatusException**

The requested operation would cause a conflict with the current state of a resource associated with the request and/or the game instance. Resolve the conflict before retrying.

HTTP Status Code: 400

**InvalidRequestException**

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

**NotFoundException**

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

**UnauthorizedException**

The client failed authentication. Clients should not retry such requests.
HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
UpdateGameSessionQueue

Updates settings for a game session queue, which determines how new game session requests in the queue are processed. To update settings, specify the queue name to be updated and provide the new settings. When updating destinations, provide a complete list of destinations.

Queue-related operations include:

- CreateGameSessionQueue (p. 37)
- DescribeGameSessionQueues (p. 120)
- UpdateGameSessionQueue (p. 250)
- DeleteGameSessionQueue (p. 73)

Request Syntax

```json
{
  "Destinations": [
    {
      "DestinationArn": "string"
    }
  ],
  "Name": "string",
  "PlayerLatencyPolicies": [
    {
      "MaximumIndividualPlayerLatencyMilliseconds": number,
      "PolicyDurationSeconds": number
    }
  ],
  "TimeoutInSeconds": number
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**

In the following list, the required parameters are described first.

**Name (p. 250)**

Descriptive label that is associated with game session queue. Queue names must be unique within each region.

Type: String


Pattern: `[a-zA-Z0-9-]+`

Required: Yes

**Destinations (p. 250)**

List of fleets that can be used to fulfill game session placement requests in the queue. Fleets are identified by either a fleet ARN or a fleet alias ARN. Destinations are listed in default preference order. When updating this list, provide a complete list of destinations.
Type: Array of GameSessionQueueDestination (p. 303) objects

Required: No

**PlayerLatencyPolicies (p. 250)**

Collection of latency policies to apply when processing game sessions placement requests with player latency information. Multiple policies are evaluated in order of the maximum latency value, starting with the lowest latency values. With just one policy, it is enforced at the start of the game session placement for the duration period. With multiple policies, each policy is enforced consecutively for its duration period. For example, a queue might enforce a 60-second policy followed by a 120-second policy, and then no policy for the remainder of the placement. When updating policies, provide a complete collection of policies.

Type: Array of PlayerLatencyPolicy (p. 324) objects

Required: No

**TimeoutInSeconds (p. 250)**

Maximum time, in seconds, that a new game session placement request remains in the queue. When a request exceeds this time, the game session placement changes to a TIMED_OUT status.

Type: Integer

Valid Range: Minimum value of 0.

Required: No

---

**Response Syntax**

```json
{
   "GameSessionQueue": {
      "Destinations": [
         {
            "DestinationArn": "string"
         }
      ],
      "GameSessionQueueArn": "string",
      "Name": "string",
      "PlayerLatencyPolicies": [
         {
            "MaximumIndividualPlayerLatencyMilliseconds": number,
            "PolicyDurationSeconds": number
         }
      ],
      "TimeoutInSeconds": number
   }
}
```

---

**Response Elements**

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**GameSessionQueue (p. 251)**

Object that describes the newly updated game session queue.
Type: GameSessionQueue (p. 301) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

InternalServiceException

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

InvalidRequestException

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

NotFoundException

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

UnauthorizedException

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
UpdateMatchmakingConfiguration

Updates settings for a FlexMatch matchmaking configuration. To update settings, specify the configuration name to be updated and provide the new settings.

Operations related to match configurations and rule sets include:

- CreateMatchmakingConfiguration (p. 41)
- DescribeMatchmakingConfigurations (p. 134)
- UpdateMatchmakingConfiguration (p. 253)
- DeleteMatchmakingConfiguration (p. 75)
- CreateMatchmakingRuleSet (p. 46)
- DescribeMatchmakingRuleSets (p. 137)
- ValidateMatchmakingRuleSet (p. 262)

Request Syntax

```
{
    "AcceptanceRequired": boolean,
    "AcceptanceTimeoutSeconds": number,
    "AdditionalPlayerCount": number,
    "CustomEventData": "string",
    "Description": "string",
    "GameProperties": [
        {
            "Key": "string",
            "Value": "string"
        }
    ],
    "GameSessionData": "string",
    "GameSessionQueueArns": [ "string" ],
    "Name": "string",
    "NotificationTarget": "string",
    "RequestTimeoutSeconds": number,
    "RuleSetName": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**
In the following list, the required parameters are described first.

**Name (p. 253)**

Unique identifier for a matchmaking configuration to update.

Type: String


Pattern: [a-zA-Z0-9-\.]+
AcceptanceRequired (p. 253)
Flag that determines whether or not a match that was created with this configuration must be accepted by the matched players. To require acceptance, set to TRUE.
Type: Boolean
Required: No
AcceptanceTimeoutSeconds (p. 253)
Length of time (in seconds) to wait for players to accept a proposed match. If any player rejects the match or fails to accept before the timeout, the ticket continues to look for an acceptable match.
Type: Integer
Required: No
AdditionalPlayerCount (p. 253)
Number of player slots in a match to keep open for future players. For example, if the configuration's rule set specifies a match for a single 12-person team, and the additional player count is set to 2, only 10 players are selected for the match.
Type: Integer
Valid Range: Minimum value of 0.
Required: No
CustomEventData (p. 253)
Information to attached to all events related to the matchmaking configuration.
Type: String
Length Constraints: Minimum length of 0. Maximum length of 256.
Required: No
Description (p. 253)
Descriptive label that is associated with matchmaking configuration.
Type: String
Required: No
GameProperties (p. 253)
Set of custom properties for a game session, formatted as key:value pairs. These properties are passed to a game server process in the GameSession (p. 290) object with a request to start a new game session (see Start a Game Session). This information is added to the new GameSession (p. 290) object that is created for a successful match.
Type: Array of GameProperty (p. 289) objects
Array Members: Maximum number of 16 items.
Required: No
GameSessionData (p. 253)

Set of custom game session properties, formatted as a single string value. This data is passed to a game server process in the GameSession (p. 290) object with a request to start a new game session (see Start a Game Session). This information is added to the new GameSession (p. 290) object that is created for a successful match.

Type: String


Required: No

GameSessionQueueArns (p. 253)

Amazon Resource Name (ARN) that is assigned to a game session queue and uniquely identifies it. Format is `arn:aws:gamelift:<region>::fleet/fleet-a1234567-b8c9-0d1e-2fa3-b45c6d7e8912`. These queues are used when placing game sessions for matches that are created with this matchmaking configuration. Queues can be located in any region.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: `[a-zA-Z0-9:/-]+`

Required: No

NotificationTarget (p. 253)

SNS topic ARN that is set up to receive matchmaking notifications. See Setting up Notifications for Matchmaking for more information.

Type: String

Length Constraints: Minimum length of 0. Maximum length of 300.

Pattern: `[a-zA-Z0-9:_/-]*`

Required: No

RequestTimeoutSeconds (p. 253)

Maximum duration, in seconds, that a matchmaking ticket can remain in process before timing out. Requests that time out can be resubmitted as needed.

Type: Integer


Required: No

RuleSetName (p. 253)

Unique identifier for a matchmaking rule set to use with this configuration. A matchmaking configuration can only use rule sets that are defined in the same region.

Type: String


Pattern: `[a-zA-Z0-9-\.]+`

Required: No
Response Syntax

```json
{
  "Configuration": {
    "AcceptanceRequired": boolean,
    "AcceptanceTimeoutSeconds": number,
    "AdditionalPlayerCount": number,
    "CreationTime": number,
    "CustomEventData": "string",
    "Description": "string",
    "GameProperties": [
      {
        "Key": "string",
        "Value": "string"
      }
    ],
    "GameSessionData": "string",
    "GameSessionQueueArns": [ "string" ],
    "Name": "string",
    "NotificationTarget": "string",
    "RequestTimeoutSeconds": number,
    "RuleSetName": "string"
  }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.

**Configuration (p. 256)**

Object that describes the updated matchmaking configuration.

Type: `MatchmakingConfiguration (p. 312)` object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

**InternalServiceException**

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

**InvalidRequestException**

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

**NotFoundException**

A service resource associated with the request could not be found. Clients should not retry such requests.
HTTP Status Code: 400

**UnsupportedRegionException**

The requested operation is not supported in the region specified.

HTTP Status Code: 400

---

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
UpdateRuntimeConfiguration

Updates the current run-time configuration for the specified fleet, which tells Amazon GameLift how to launch server processes on instances in the fleet. You can update a fleet's run-time configuration at any time after the fleet is created; it does not need to be in an ACTIVE status.

To update run-time configuration, specify the fleet ID and provide a RuntimeConfiguration object with the updated collection of server process configurations.

Each instance in a Amazon GameLift fleet checks regularly for an updated run-time configuration and changes how it launches server processes to comply with the latest version. Existing server processes are not affected by the update; they continue to run until they end, while Amazon GameLift simply adds new server processes to fit the current run-time configuration. As a result, the run-time configuration changes are applied gradually as existing processes shut down and new processes are launched in Amazon GameLift's normal process recycling activity.

Fleet-related operations include:

- CreateFleet (p. 19)
- ListFleets (p. 173)
- DeleteFleet (p. 70)
- Describe fleets:
  - DescribeFleetAttributes (p. 93)
  - DescribeFleetCapacity (p. 97)
  - DescribeFleetPortSettings (p. 107)
  - DescribeFleetUtilization (p. 110)
  - DescribeRuntimeConfiguration (p. 144)
  - DescribeEC2InstanceLimits (p. 90)
  - DescribeFleetEvents (p. 101)
- Update fleets:
  - UpdateFleetAttributes (p. 234)
  - UpdateFleetCapacity (p. 238)
  - UpdateFleetPortSettings (p. 242)
  - UpdateRuntimeConfiguration (p. 258)
- Manage fleet actions:
  - StartFleetActions (p. 201)
  - StopFleetActions (p. 219)

Request Syntax

```json
{
    "FleetId": "string",
    "RuntimeConfiguration": {
        "GameSessionActivationTimeoutSeconds": number,
        "MaxConcurrentGameSessionActivations": number,
        "ServerProcesses": [
            {
                "ConcurrentExecutions": number,
                "LaunchPath": "string",
                "Parameters": "string"
            }
        ]
    }
}
```

API Version 2015-10-01
Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**
In the following list, the required parameters are described first.

**FleetId (p. 258)**

Unique identifier for a fleet to update run-time configuration for.

Type: String

Pattern: ^fleet-\S+

Required: Yes

**RuntimeConfiguration (p. 258)**

Instructions for launching server processes on each instance in the fleet. The run-time configuration for a fleet has a collection of server process configurations, one for each type of server process to run on an instance. A server process configuration specifies the location of the server executable, launch parameters, and the number of concurrent processes with that configuration to maintain on each instance.

Type: RuntimeConfiguration (p. 331) object

Required: Yes

Response Syntax

```
{
  "RuntimeConfiguration": {
    "GameSessionActivationTimeoutSeconds": number,
    "MaxConcurrentGameSessionActivations": number,
    "ServerProcesses": [
      {
        "ConcurrentExecutions": number,
        "LaunchPath": "string",
        "Parameters": "string"
      }
    ]
  }
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.

The following data is returned in JSON format by the service.
RuntimeConfiguration (p. 259)

The run-time configuration currently in force. If the update was successful, this object matches the one in the request.

Type: RuntimeConfiguration (p. 331) object

Errors

For information about the errors that are common to all actions, see Common Errors (p. 347).

InternalServiceException

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

InvalidFleetStatusException

The requested operation would cause a conflict with the current state of a resource associated with the request and/or the fleet. Resolve the conflict before retrying.

HTTP Status Code: 400

InvalidRequestException

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

NotFoundException

A service resource associated with the request could not be found. Clients should not retry such requests.

HTTP Status Code: 400

UnauthorizedException

The client failed authentication. Clients should not retry such requests.

HTTP Status Code: 400

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
ValidateMatchmakingRuleSet

Validates the syntax of a matchmaking rule or rule set. This operation checks that the rule set uses syntactically correct JSON and that it conforms to allowed property expressions. To validate syntax, provide a rule set string.

Operations related to match configurations and rule sets include:

- CreateMatchmakingConfiguration (p. 41)
- DescribeMatchmakingConfigurations (p. 134)
- UpdateMatchmakingConfiguration (p. 253)
- DeleteMatchmakingConfiguration (p. 75)
- CreateMatchmakingRuleSet (p. 46)
- DescribeMatchmakingRuleSets (p. 137)
- ValidateMatchmakingRuleSet (p. 262)

Request Syntax

```json
{
   "RuleSetBody": "string"
}
```

Request Parameters

For information about the parameters that are common to all actions, see Common Parameters (p. 345).

The request accepts the following data in JSON format.

**Note**
In the following list, the required parameters are described first.

**RuleSetBody (p. 262)**
Collection of matchmaking rules to validate, formatted as a JSON string.

Type: String


Required: Yes

Response Syntax

```json
{
   "Valid": boolean
}
```

Response Elements

If the action is successful, the service sends back an HTTP 200 response.
The following data is returned in JSON format by the service.

**Valid (p. 262)**

Response indicating whether or not the rule set is valid.

Type: Boolean

**Errors**

For information about the errors that are common to all actions, see Common Errors (p. 347).

**InternalServiceException**

The service encountered an unrecoverable internal failure while processing the request. Clients can retry such requests immediately or after a waiting period.

HTTP Status Code: 500

**InvalidRequestException**

One or more parameter values in the request are invalid. Correct the invalid parameter values before retrying.

HTTP Status Code: 400

**UnsupportedRegionException**

The requested operation is not supported in the region specified.

HTTP Status Code: 400

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS Command Line Interface
- AWS SDK for .NET
- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for JavaScript
- AWS SDK for PHP V3
- AWS SDK for Python
- AWS SDK for Ruby V2
Data Types

The Amazon GameLift API contains several data types that various actions use. This section describes each data type in detail.

**Note**
The order of each element in a data type structure is not guaranteed. Applications should not assume a particular order.

The following data types are supported:

- Alias (p. 266)
- AttributeValue (p. 268)
- AwsCredentials (p. 270)
- Build (p. 271)
- DesiredPlayerSession (p. 273)
- EC2InstanceCounts (p. 274)
- EC2InstanceLimit (p. 276)
- Event (p. 277)
- FleetAttributes (p. 280)
- FleetCapacity (p. 285)
- FleetUtilization (p. 287)
- GameProperty (p. 289)
- GameSession (p. 290)
- GameSessionConnectionInfo (p. 294)
- GameSessionDetail (p. 296)
- GameSessionPlacement (p. 297)
- GameSessionQueue (p. 301)
- GameSessionQueueDestination (p. 303)
- Instance (p. 304)
- InstanceAccess (p. 306)
- InstanceCredentials (p. 308)
- IpPermission (p. 309)
- MatchedPlayerSession (p. 311)
- MatchmakingConfiguration (p. 312)
- MatchmakingRuleSet (p. 315)
- MatchmakingTicket (p. 317)
- PlacedPlayerSession (p. 320)
- Player (p. 321)
- PlayerLatency (p. 323)
- PlayerLatencyPolicy (p. 324)
- PlayerSession (p. 325)
- ResourceCreationLimitPolicy (p. 328)
- RoutingStrategy (p. 329)
- RuntimeConfiguration (p. 331)
- S3Location (p. 333)
• ScalingPolicy (p. 334)
• ServerProcess (p. 338)
• TargetConfiguration (p. 339)
• VpcPeeringAuthorization (p. 340)
• VpcPeeringConnection (p. 342)
• VpcPeeringConnectionStatus (p. 344)
Alias

Properties describing a fleet alias.

Alias-related operations include:

- `CreateAlias (p. 10)`
- `ListAliases (p. 165)`
- `DescribeAlias (p. 84)`
- `UpdateAlias (p. 227)`
- `DeleteAlias (p. 65)`
- `ResolveAlias (p. 189)`

Contents

**Note**

In the following list, the required parameters are described first.

**AliasArn**

Unique identifier for an alias; alias ARNs are unique across all regions.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: `[^a-zA-Z0-9:/-]+`

Required: No

**AliasId**

Unique identifier for an alias; alias IDs are unique within a region.

Type: String

Pattern: `^alias-\S+$`

Required: No

**CreationTime**

Time stamp indicating when this data object was created. Format is a number expressed in Unix time as milliseconds (for example "1469498468.057")

Type: Timestamp

Required: No

**Description**

Human-readable description of an alias.

Type: String

Required: No

**LastUpdatedTime**

Time stamp indicating when this data object was last modified. Format is a number expressed in Unix time as milliseconds (for example "1469498468.057").
Type: Timestamp
Required: No

Name

Descriptive label that is associated with an alias. Alias names do not need to be unique.

Type: String


Pattern: .+\S+.+

Required: No

RoutingStrategy

Alias configuration for the alias, including routing type and settings.

Type: RoutingStrategy (p. 329) object

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2
AttributeValue

Values for use in Player (p. 321) attribute key:value pairs. This object lets you specify an attribute value using any of the valid data types: string, number, string array or data map. Each AttributeValue object can use only one of the available properties.

Contents

Note
In the following list, the required parameters are described first.

N
For number values, expressed as double.
Type: Double
Required: No

S
For single string values. Maximum string length is 100 characters.
Type: String
Required: No

SDM
For a map of up to 10 data type:value pairs. Maximum length for each string value is 100 characters.
Type: String to double map
Key Length Constraints: Minimum length of 1. Maximum length of 1024.
Required: No

SL
For a list of up to 10 strings. Maximum length for each string is 100 characters. Duplicate values are not recognized; all occurrences of the repeated value after the first of a repeated value are ignored.
Type: Array of strings
Required: No

See Also
For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2
AwsCredentials

Temporary access credentials used for uploading game build files to Amazon GameLift. They are valid for a limited time. If they expire before you upload your game build, get a new set by calling RequestUploadCredentials (p. 186).

Contents

Note
In the following list, the required parameters are described first.

AccessKeyId
Temporary key allowing access to the Amazon GameLift S3 account.
Type: String
Length Constraints: Minimum length of 1.
Required: No

SecretAccessKey
Temporary secret key allowing access to the Amazon GameLift S3 account.
Type: String
Length Constraints: Minimum length of 1.
Required: No

SessionToken
Token used to associate a specific build ID with the files uploaded using these credentials.
Type: String
Length Constraints: Minimum length of 1.
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2
Build

Properties describing a game build.

Build-related operations include:

- CreateBuild (p. 13)
- ListBuilds (p. 168)
- DescribeBuild (p. 87)
- UpdateBuild (p. 230)
- DeleteBuild (p. 67)

Contents

Note
In the following list, the required parameters are described first.

BuildId

Unique identifier for a build.

Type: String

Pattern: ^build-\S+

Required: No

CreationTime

Time stamp indicating when this data object was created. Format is a number expressed in Unix time as milliseconds (for example "1469498468.057").

Type: Timestamp

Required: No

Name

Descriptive label that is associated with a build. Build names do not need to be unique. It can be set using CreateBuild (p. 13) or UpdateBuild (p. 230).

Type: String

Required: No

OperatingSystem

Operating system that the game server binaries are built to run on. This value determines the type of fleet resources that you can use for this build.

Type: String

Valid Values: WINDOWS_2012 | AMAZON_LINUX

Required: No

SizeOnDisk

File size of the uploaded game build, expressed in bytes. When the build status is INITIALIZED, this value is 0.
Type: Long
Valid Range: Minimum value of 1.
Required: No

**Status**

Current status of the build.

Possible build status includes the following:
- **INITIALIZED** -- A new build has been defined, but no files have been uploaded. You cannot create fleets for builds that are in this status. When a build is successfully created, the build status is set to this value.
- **READY** -- The game build has been successfully uploaded. You can now create new fleets for this build.
- **FAILED** -- The game build upload failed. You cannot create new fleets for this build.

Type: String
Valid Values: INITIALIZED | READY | FAILED
Required: No

**Version**

Version that is associated with this build. Version strings do not need to be unique. This value can be set using CreateBuild (p. 13) or UpdateBuild (p. 230).

Type: String
Required: No

### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2
**DesiredPlayerSession**

Player information for use when creating player sessions using a game session placement request with `StartGameSessionPlacement (p. 204)`.

**Contents**

**Note**
In the following list, the required parameters are described first.

**PlayerData**

Developer-defined information related to a player. Amazon GameLift does not use this data, so it can be formatted as needed for use in the game.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Required: No

**PlayerId**

Unique identifier for a player to associate with the player session.

Type: String


Required: No

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2
EC2InstanceCounts

Current status of fleet capacity. The number of active instances should match or be in the process of matching the number of desired instances. Pending and terminating counts are non-zero only if fleet capacity is adjusting to an UpdateFleetCapacity (p. 238) request, or if access to resources is temporarily affected.

Fleet-related operations include:

- CreateFleet (p. 19)
- ListFleets (p. 173)
- DeleteFleet (p. 70)
- Describe fleets:
  - DescribeFleetAttributes (p. 93)
  - DescribeFleetCapacity (p. 97)
  - DescribeFleetPortSettings (p. 107)
  - DescribeFleetUtilization (p. 110)
  - DescribeRuntimeConfiguration (p. 144)
  - DescribeEC2InstanceLimits (p. 90)
  - DescribeFleetEvents (p. 101)
- Update fleets:
  - UpdateFleetAttributes (p. 234)
  - UpdateFleetCapacity (p. 238)
  - UpdateFleetPortSettings (p. 242)
  - UpdateRuntimeConfiguration (p. 258)
- Manage fleet actions:
  - StartFleetActions (p. 201)
  - StopFleetActions (p. 219)

Contents

Note
In the following list, the required parameters are described first.

ACTIVE

Actual number of active instances in the fleet.

Type: Integer

Valid Range: Minimum value of 0.

Required: No

DESIRED

Ideal number of active instances in the fleet.

Type: Integer

Valid Range: Minimum value of 0.

Required: No
IDLE
Number of active instances in the fleet that are not currently hosting a game session.
Type: Integer
Valid Range: Minimum value of 0.
Required: No

MAXIMUM
Maximum value allowed for the fleet's instance count.
Type: Integer
Valid Range: Minimum value of 0.
Required: No

MINIMUM
Minimum value allowed for the fleet's instance count.
Type: Integer
Valid Range: Minimum value of 0.
Required: No

PENDING
Number of instances in the fleet that are starting but not yet active.
Type: Integer
Valid Range: Minimum value of 0.
Required: No

TERMINATING
Number of instances in the fleet that are no longer active but haven't yet been terminated.
Type: Integer
Valid Range: Minimum value of 0.
Required: No

See Also
For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2
**EC2InstanceLimit**

Maximum number of instances allowed based on the Amazon Elastic Compute Cloud (Amazon EC2) instance type. Instance limits can be retrieved by calling DescribeEC2InstanceLimits (p. 90).

**Contents**

**Note**

In the following list, the required parameters are described first.

**CurrentInstances**

Number of instances of the specified type that are currently in use by this AWS account.

Type: Integer

Valid Range: Minimum value of 0.

Required: No

**EC2InstanceType**

Name of an EC2 instance type that is supported in Amazon GameLift. A fleet instance type determines the computing resources of each instance in the fleet, including CPU, memory, storage, and networking capacity. Amazon GameLift supports the following EC2 instance types. See Amazon EC2 Instance Types for detailed descriptions.

Type: String

Valid Values: t2.micro | t2.small | t2.medium | t2.large | c3.large | c3.xlarge | c3.2xlarge | c3.4xlarge | c3.8xlarge | c4.large | c4.xlarge | c4.2xlarge | c4.4xlarge | c4.8xlarge | r3.large | r3.xlarge | r3.2xlarge | r3.4xlarge | r4.large | r4.xlarge | r4.2xlarge | r4.4xlarge | r4.8xlarge | r4.16xlarge | m3.medium | m3.large | m3.xlarge | m3.2xlarge | m4.large | m4.xlarge | m4.2xlarge | m4.4xlarge | m4.10xlarge

Required: No

**InstanceLimit**

Number of instances allowed.

Type: Integer

Valid Range: Minimum value of 0.

Required: No

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2
Event

Log entry describing an event that involves Amazon GameLift resources (such as a fleet). In addition to tracking activity, event codes and messages can provide additional information for troubleshooting and debugging problems.

Contents

Note
In the following list, the required parameters are described first.

EventCode

Type of event being logged. The following events are currently in use:

Fleet creation events:
• FLEET_CREATED -- A fleet record was successfully created with a status of NEW. Event messaging includes the fleet ID.
• FLEET_STATE_DOWNLOADING -- Fleet status changed from NEW to DOWNLOADING. The compressed build has started downloading to a fleet instance for installation.
• FLEET_BINARY_DOWNLOAD_FAILED -- The build failed to download to the fleet instance.
• FLEET_CREATION_EXTRACTING_BUILD – The game server build was successfully downloaded to an instance, and the build files are now being extracted from the uploaded build and saved to an instance. Failure at this stage prevents a fleet from moving to ACTIVE status. Logs for this stage display a list of the files that are extracted and saved on the instance. Access the logs by using the URL in PreSignedLogUrl.
• FLEET_CREATION_RUNNING_INSTALLER – The game server build files were successfully extracted, and the Amazon GameLift is now running the build's install script (if one is included). Failure in this stage prevents a fleet from moving to ACTIVE status. Logs for this stage list the installation steps and whether or not the install completed successfully. Access the logs by using the URL in PreSignedLogUrl.
• FLEET_CREATION_VALIDATING_RUNTIME_CONFIG -- The build process was successful, and the Amazon GameLift is now verifying that the game server launch paths, which are specified in the fleet's run-time configuration, exist. If any listed launch path exists, Amazon GameLift tries to launch a game server process and waits for the process to report ready. Failures in this stage prevent a fleet from moving to ACTIVE status. Logs for this stage list the launch paths in the run-time configuration and indicate whether each is found. Access the logs by using the URL in PreSignedLogUrl.
• FLEET_STATE_VALIDATING -- Fleet status changed from DOWNLOADING to VALIDATING.
• FLEET_VALIDATION_LAUNCH_PATH_NOT_FOUND -- Validation of the run-time configuration failed because the executable specified in a launch path does not exist on the instance.
• FLEET_STATE_BUILDING -- Fleet status changed from VALIDATING to BUILDING.
• FLEET_VALIDATION_EXECUTABLE_RUNTIME_FAILURE -- Validation of the run-time configuration failed because the executable specified in a launch path failed to run on the fleet instance.
• FLEET_STATE_ACTIVATING -- Fleet status changed from BUILDING to ACTIVATING.
• FLEET_ACTIVATION_FAILED - The fleet failed to successfully complete one of the steps in the fleet activation process. This event code indicates that the game build was successfully downloaded to a fleet instance, built, and validated, but was not able to start a server process. A possible reason for failure is that the game server is not reporting "process ready" to the Amazon GameLift service.
• FLEET_STATE_ACTIVE -- The fleet's status changed from ACTIVATING to ACTIVE. The fleet is now ready to host game sessions.

VPC peering events:
• **FLEET_VPC_PEERING_SUCCEEDED** -- A VPC peering connection has been established between the VPC for an Amazon GameLift fleet and a VPC in your AWS account.

• **FLEET_VPC_PEERING_FAILED** -- A requested VPC peering connection has failed. Event details and status information (see DescribeVpcPeeringConnections (p. 154)) provide additional detail. A common reason for peering failure is that the two VPCs have overlapping CIDR blocks of IPv4 addresses. To resolve this, change the CIDR block for the VPC in your AWS account. For more information on VPC peering failures, see https://docs.aws.amazon.com/AmazonVPC/latest/PeeringGuide/invalid-peering-configurations.html

• **FLEET_VPC_PEERING_DELETED** -- A VPC peering connection has been successfully deleted.

**Spot instance events:**

• **INSTANCE_INTERRUPTED** -- A spot instance was interrupted by EC2 with a two-minute notification.

**Other fleet events:**

• **FLEET_SCALING_EVENT** -- A change was made to the fleet's capacity settings (desired instances, minimum/maximum scaling limits). Event messaging includes the new capacity settings.

• **FLEET_NEW_GAME_SESSION_PROTECTION_POLICY_UPDATED** -- A change was made to the fleet's game session protection policy setting. Event messaging includes both the old and new policy setting.

• **FLEET_DELETED** -- A request to delete a fleet was initiated.

• **GENERIC_EVENT** -- An unspecified event has occurred.

**Type:** String

**Valid Values:** GENERIC_EVENT | FLEET_CREATED | FLEET_DELETED | FLEET_SCALING_EVENT | FLEET_STATE_DOWNLOADING | FLEET_STATE_VALIDATING | FLEET_STATE_BUILDING | FLEET_STATE_ACTIVATING | FLEET_STATE_ACTIVE | FLEET_STATE_ERROR | FLEET_INITIALIZATION_FAILED | FLEET_BINARY_DOWNLOAD_FAILED | FLEET_VALIDATION_LAUNCH_PATH_NOT_FOUND | FLEET_VALIDATION_EXECUTABLE_RUNTIME_FAILURE | FLEET_VALIDATION_TIMED_OUT | FLEET_ACTIVATION_FAILED | FLEET_ACTIVATION_FAILED_NO_INSTANCES | FLEET_NEW_GAME_SESSION_PROTECTION_POLICY_UPDATED | SERVER_PROCESS_INVALID_PATH | SERVER_PROCESS_SDK_INITIALIZATION_TIMEOUT | SERVER_PROCESS_PROCESS_READY_TIMEOUT | SERVER_PROCESS_CRASHED | SERVER_PROCESS_TERMINATED_UNHEALTHY | SERVER_PROCESS_FORCE_TERMINATED | SERVER_PROCESS_PROCESS_EXIT_TIMEOUT | GAME_SESSION_ACTIVATION_TIMEOUT | FLEET_CREATION_EXTRACTING_BUILD | FLEET_CREATION_RUNNING_INSTALLER | FLEET_CREATION_VALIDATING_RUNTIME_CONFIG | FLEET_VPC_PEERING_SUCCEEDED | FLEET_VPC_PEERING_FAILED | FLEET_VPC_PEERING_DELETED | INSTANCE_INTERRUPTED

**Required:** No

**EventId**

Unique identifier for a fleet event.

**Type:** String

**Length Constraints:** Minimum length of 1. Maximum length of 1024.

**Required:** No

**EventTime**

Time stamp indicating when this event occurred. Format is a number expressed in Unix time as milliseconds (for example "1469498468.057").

**Type:** Timestamp
Required: No

**Message**

Additional information related to the event.

Type: String

Length Constraints: Minimum length of 1.

Required: No

**PreSignedLogUrl**

Location of stored logs with additional detail that is related to the event. This is useful for debugging issues. The URL is valid for 15 minutes. You can also access fleet creation logs through the Amazon GameLift console.

Type: String


Required: No

**ResourceId**

Unique identifier for an event resource, such as a fleet ID.

Type: String


Required: No

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2
FleetAttributes

General properties describing a fleet.

Fleet-related operations include:

- CreateFleet (p. 19)
- ListFleets (p. 173)
- DeleteFleet (p. 70)
- Describe fleets:
  - DescribeFleetAttributes (p. 93)
  - DescribeFleetCapacity (p. 97)
  - DescribeFleetPortSettings (p. 107)
  - DescribeFleetUtilization (p. 110)
  - DescribeRuntimeConfiguration (p. 144)
  - DescribeEC2InstanceLimits (p. 90)
  - DescribeFleetEvents (p. 101)
- Update fleets:
  - UpdateFleetAttributes (p. 234)
  - UpdateFleetCapacity (p. 238)
  - UpdateFleetPortSettings (p. 242)
  - UpdateRuntimeConfiguration (p. 258)
- Manage fleet actions:
  - StartFleetActions (p. 201)
  - StopFleetActions (p. 219)

Contents

Note
In the following list, the required parameters are described first.

BuildId

Unique identifier for a build.

Type: String

Pattern: ^build-[\S]+$

Required: No

CreationTime

Time stamp indicating when this data object was created. Format is a number expressed in Unix time as milliseconds (for example "1469498468.057").

Type: Timestamp

Required: No

Description

Human-readable description of the fleet.

Type: String

Required: No

**FleetArn**

Identifier for a fleet that is unique across all regions.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: [a-zA-Z0-9:/-]+  

Required: No

**FleetId**

Unique identifier for a fleet.

Type: String

Pattern: ^fleet-\S+

Required: No

**FleetType**

Indicates whether the fleet uses on-demand or spot instances. A spot instance in use may be interrupted with a two-minute notification.

Type: String

Valid Values: ON_DEMAND | SPOT

Required: No

**InstanceType**

EC2 instance type indicating the computing resources of each instance in the fleet, including CPU, memory, storage, and networking capacity. See Amazon EC2 Instance Types for detailed descriptions.

Type: String

Valid Values: t2.micro | t2.small | t2.medium | t2.large | c3.large | c3.xlarge | c3.2xlarge | c3.4xlarge | c3.8xlarge | c4.large | c4.xlarge | c4.2xlarge | c4.4xlarge | c4.8xlarge | r3.large | r3.xlarge | r3.2xlarge | r3.8xlarge | r4.large | r4.xlarge | r4.2xlarge | r4.4xlarge | r4.8xlarge | r4.16xlarge | m3.medium | m3.large | m3.xlarge | m3.2xlarge | m4.large | m4.xlarge | m4.2xlarge | m4.4xlarge | m4.10xlarge

Required: No

**LogPaths**

Location of default log files. When a server process is shut down, Amazon GameLift captures and stores any log files in this location. These logs are in addition to game session logs; see more on game session logs in the Amazon GameLift Developer Guide. If no default log path for a fleet is specified, Amazon GameLift automatically uploads logs that are stored on each instance at C:\game \logs (for Windows) or /local/game/logs (for Linux). Use the Amazon GameLift console to access stored logs.

Type: Array of strings

Required: No

**MetricGroups**

Names of metric groups that this fleet is included in. In Amazon CloudWatch, you can view metrics for an individual fleet or aggregated metrics for fleets that are in a fleet metric group. A fleet can be included in only one metric group at a time.

Type: Array of strings

Array Members: Maximum number of 1 item.

Length Constraints: Minimum length of 1. Maximum length of 255.

Required: No

**Name**

Descriptive label that is associated with a fleet. Fleet names do not need to be unique.

Type: String


Required: No

**NewGameSessionProtectionPolicy**

Type of game session protection to set for all new instances started in the fleet.

- **NoProtection** -- The game session can be terminated during a scale-down event.
- **FullProtection** -- If the game session is in an **ACTIVE** status, it cannot be terminated during a scale-down event.

Type: String

Valid Values: NoProtection | FullProtection

Required: No

**OperatingSystem**

Operating system of the fleet's computing resources. A fleet's operating system depends on the OS specified for the build that is deployed on this fleet.

Type: String

Valid Values: WINDOWS_2012 | AMAZON_LINUX

Required: No

**ResourceCreationLimitPolicy**

Fleet policy to limit the number of game sessions an individual player can create over a span of time.

Type: ResourceCreationLimitPolicy (p. 328) object

Required: No

**ServerLaunchParameters**

Game server launch parameters specified for fleets created before 2016-08-04 (or AWS SDK v. 0.12.16). Server launch parameters for fleets created after this date are specified in the fleet's RuntimeConfiguration (p. 331).

Type: String

**NewGameSessionProtectionPolicy**

Type of game session protection to set for all new instances started in the fleet.

- **NoProtection** -- The game session can be terminated during a scale-down event.
- **FullProtection** -- If the game session is in an **ACTIVE** status, it cannot be terminated during a scale-down event.

Type: String

Valid Values: NoProtection | FullProtection

Required: No

**OperatingSystem**

Operating system of the fleet's computing resources. A fleet's operating system depends on the OS specified for the build that is deployed on this fleet.

Type: String

Valid Values: WINDOWS_2012 | AMAZON_LINUX

Required: No

**ResourceCreationLimitPolicy**

Fleet policy to limit the number of game sessions an individual player can create over a span of time.

Type: ResourceCreationLimitPolicy (p. 328) object

Required: No

**ServerLaunchParameters**

Game server launch parameters specified for fleets created before 2016-08-04 (or AWS SDK v. 0.12.16). Server launch parameters for fleets created after this date are specified in the fleet's RuntimeConfiguration (p. 331).

Type: String

Required: No

**ServerLaunchPath**

Path to a game server executable in the fleet's build, specified for fleets created before 2016-08-04 (or AWS SDK v. 0.12.16). Server launch paths for fleets created after this date are specified in the fleet's `RuntimeConfiguration (p. 331)`.

Type: String


Required: No

**Status**

Current status of the fleet.

Possible fleet statuses include the following:

- **NEW** -- A new fleet has been defined and desired instances is set to 1.
- **DOWNLOADING/VALIDATING/BUILDING/ACTIVATING** -- Amazon GameLift is setting up the new fleet, creating new instances with the game build and starting server processes.
- **ACTIVE** -- Hosts can now accept game sessions.
- **ERROR** -- An error occurred when downloading, validating, building, or activating the fleet.
- **DELETING** -- Hosts are responding to a delete fleet request.
- **TERMINATED** -- The fleet no longer exists.

Type: String

Valid Values: `NEW` | `DOWNLOADING` | `VALIDATING` | `BUILDING` | `ACTIVATING` | `ACTIVE` | `DELETING` | `ERROR` | `TERMINATED`

Required: No

**StoppedActions**

List of fleet actions that have been suspended using `StopFleetActions (p. 219)`. This includes auto-scaling.

Type: Array of strings

Array Members: Fixed number of 1 item.

Valid Values: `AUTO_SCALING`

Required: No

**TerminationTime**

Time stamp indicating when this data object was terminated. Format is a number expressed in Unix time as milliseconds (for example "1469498468.057").

Type: Timestamp

Required: No

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:
• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java
• AWS SDK for Ruby V2
FleetCapacity

Information about the fleet's capacity. Fleet capacity is measured in EC2 instances. By default, new fleets have a capacity of one instance, but can be updated as needed. The maximum number of instances for a fleet is determined by the fleet's instance type.

Fleet-related operations include:

- CreateFleet (p. 19)
- ListFleets (p. 173)
- DeleteFleet (p. 70)
- Describe fleets:
  - DescribeFleetAttributes (p. 93)
  - DescribeFleetCapacity (p. 97)
  - DescribeFleetPortSettings (p. 107)
  - DescribeFleetUtilization (p. 110)
  - DescribeRuntimeConfiguration (p. 144)
  - DescribeEC2InstanceLimits (p. 90)
  - DescribeFleetEvents (p. 101)
- Update fleets:
  - UpdateFleetAttributes (p. 234)
  - UpdateFleetCapacity (p. 238)
  - UpdateFleetPortSettings (p. 242)
  - UpdateRuntimeConfiguration (p. 258)
- Manage fleet actions:
  - StartFleetActions (p. 201)
  - StopFleetActions (p. 219)

Contents

Note
In the following list, the required parameters are described first.

FleetId

Unique identifier for a fleet.

Type: String

Pattern: \^fleet-\S+

Required: No

InstanceCounts

Current status of fleet capacity.

Type: EC2InstanceCounts (p. 274) object

Required: No

InstanceType

Name of an EC2 instance type that is supported in Amazon GameLift. A fleet instance type determines the computing resources of each instance in the fleet, including CPU, memory, storage,
and networking capacity. Amazon GameLift supports the following EC2 instance types. See Amazon
EC2 Instance Types for detailed descriptions.

**Type:** String

**Valid Values:**
- t2.micro
- t2.small
- t2.medium
- t2.large
- c3.large
- c3.xlarge
- c3.2xlarge
- c3.4xlarge
- c3.8xlarge
- c4.large
- c4.xlarge
- c4.2xlarge
- c4.4xlarge
- c4.8xlarge
- r3.large
- r3.xlarge
- r3.2xlarge
- r3.4xlarge
- r4.large
- r4.xlarge
- r4.2xlarge
- r4.4xlarge
- r4.8xlarge
- r4.16xlarge
- m3.medium
- m3.large
- m3.xlarge
- m3.2xlarge
- m4.large
- m4.xlarge
- m4.2xlarge
- m4.4xlarge
- m4.10xlarge

**Required:** No

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2
FleetUtilization

Current status of fleet utilization, including the number of game and player sessions being hosted.

Fleet-related operations include:

- CreateFleet (p. 19)
- ListFleets (p. 173)
- DeleteFleet (p. 70)
- Describe fleets:
  - DescribeFleetAttributes (p. 93)
  - DescribeFleetCapacity (p. 97)
  - DescribeFleetPortSettings (p. 107)
  - DescribeFleetUtilization (p. 110)
  - DescribeRuntimeConfiguration (p. 144)
  - DescribeEC2InstanceLimits (p. 90)
  - DescribeFleetEvents (p. 101)
- Update fleets:
  - UpdateFleetAttributes (p. 234)
  - UpdateFleetCapacity (p. 238)
  - UpdateFleetPortSettings (p. 242)
  - UpdateRuntimeConfiguration (p. 258)
- Manage fleet actions:
  - StartFleetActions (p. 201)
  - StopFleetActions (p. 219)

Contents

**Note**
In the following list, the required parameters are described first.

**ActiveGameSessionCount**

Number of active game sessions currently being hosted on all instances in the fleet.

Type: Integer

Valid Range: Minimum value of 0.

Required: No

**ActiveServerProcessCount**

Number of server processes in an ACTIVE status currently running across all instances in the fleet

Type: Integer

Valid Range: Minimum value of 0.

Required: No

**CurrentPlayerSessionCount**

Number of active player sessions currently being hosted on all instances in the fleet.
See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2
GameProperty

Set of key-value pairs that contain information about a game session. When included in a game session request, these properties communicate details to be used when setting up the new game session, such as to specify a game mode, level, or map. Game properties are passed to the game server process when initiating a new game session; the server process uses the properties as appropriate. For more information, see the Amazon GameLift Developer Guide.

Contents

**Note**
In the following list, the required parameters are described first.

**Key**
Game property identifier.
Type: String
Length Constraints: Maximum length of 32.
Required: Yes

**Value**
Game property value.
Type: String
Length Constraints: Maximum length of 96.
Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2
GameSession

Properties describing a game session.

A game session in ACTIVE status can host players. When a game session ends, its status is set to TERMINATED.

Once the session ends, the game session object is retained for 30 days. This means you can reuse idempotency token values after this time. Game session logs are retained for 14 days.

Game-session-related operations include:

- CreateGameSession (p. 31)
- DescribeGameSessions (p. 123)
- DescribeGameSessionDetails (p. 113)
- SearchGameSessions (p. 192)
- UpdateGameSession (p. 246)
- GetGameSessionLogUrl (p. 157)
- Game session placements
  - StartGameSessionPlacement (p. 204)
  - DescribeGameSessionPlacement (p. 117)
  - StopGameSessionPlacement (p. 222)

Contents

Note
In the following list, the required parameters are described first.

CreationTime

Time stamp indicating when this data object was created. Format is a number expressed in Unix time as milliseconds (for example "1469498468.057").

Type: Timestamp

Required: No

CreatorId

Unique identifier for a player. This ID is used to enforce a resource protection policy (if one exists), that limits the number of game sessions a player can create.

Type: String


Required: No

CurrentPlayerSessionCount

Number of players currently in the game session.

Type: Integer

Valid Range: Minimum value of 0.

Required: No
FleetId

Unique identifier for a fleet that the game session is running on.

Type: String

Pattern: ^fleet-\S+

Required: No

GameProperties

Set of custom properties for a game session, formatted as key:value pairs. These properties are passed to a game server process in the GameSession (p. 290) object with a request to start a new game session (see Start a Game Session). You can search for active game sessions based on this custom data with SearchGameSessions (p. 192).

Type: Array of GameProperty (p. 289) objects

Array Members: Maximum number of 16 items.

Required: No

GameSessionData

Set of custom game session properties, formatted as a single string value. This data is passed to a game server process in the GameSession (p. 290) object with a request to start a new game session (see Start a Game Session).

Type: String


Required: No

GameSessionId

Unique identifier for the game session. A game session ARN has the following format: arn:aws:gamelift:<region>::gamesession/<fleet ID>/<custom ID string or idempotency token>.

Type: String


Required: No

IpAddress

IP address of the game session. To connect to a Amazon GameLift game server, an app needs both the IP address and port number.

Type: String

Required: No

MatchmakerData

Information about the matchmaking process that was used to create the game session. It is in JSON syntax, formatted as a string. In addition the matchmaking configuration used, it contains data on all players assigned to the match, including player attributes and team assignments. For more details on matchmaking data, see Match Data. Matchmaker data is useful when requesting match backfills, and is updated whenever new players are added during a successful backfill (see StartMatchBackfill (p. 210)).
**Type**: String


Required: No

**MaximumPlayerSessionCount**

Maximum number of players that can be connected simultaneously to the game session.

Type: Integer

Valid Range: Minimum value of 0.

Required: No

**Name**

Descriptive label that is associated with a game session. Session names do not need to be unique.

Type: String


Required: No

**PlayerSessionCreationPolicy**

Indicates whether or not the game session is accepting new players.

Type: String

Valid Values: ACCEPT_ALL | DENY_ALL

Required: No

**Port**

Port number for the game session. To connect to a Amazon GameLift game server, an app needs both the IP address and port number.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 60000.

Required: No

**Status**

Current status of the game session. A game session must have an ACTIVE status to have player sessions.

Type: String

Valid Values: ACTIVE | ACTIVATING | TERMINATED | TERMINATING | ERROR

Required: No

**StatusReason**

Provides additional information about game session status. INTERRUPTED indicates that the game session was hosted on a spot instance that was reclaimed, causing the active game session to be terminated.

Type: String
Valid Values: INTERRUPTED

Required: No

**TerminationTime**

Time stamp indicating when this data object was terminated. Format is a number expressed in Unix time as milliseconds (for example "1469498468.057").

Type: Timestamp

Required: No

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2
GameSessionConnectionInfo

Connection information for the new game session that is created with matchmaking. (with StartMatchmaking (p. 214)). Once a match is set, the FlexMatch engine places the match and creates a new game session for it. This information, including the game session endpoint and player sessions for each player in the original matchmaking request, is added to the MatchmakingTicket (p. 317), which can be retrieved by calling DescribeMatchmaking (p. 131).

Contents

Note
In the following list, the required parameters are described first.

GameSessionArn

Amazon Resource Name (ARN) that is assigned to a game session and uniquely identifies it.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: [a-zA-Z0-9:/-]+

Required: No

IpAddress

IP address of the game session. To connect to a Amazon GameLift game server, an app needs both the IP address and port number.

Type: String

Required: No

MatchedPlayerSessions

Collection of player session IDs, one for each player ID that was included in the original matchmaking request.

Type: Array of MatchedPlayerSession (p. 311) objects

Required: No

Port

Port number for the game session. To connect to a Amazon GameLift game server, an app needs both the IP address and port number.

Type: Integer

Valid Range: Minimum value of 1.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java
• AWS SDK for Ruby V2
GameSessionDetail

A game session's properties plus the protection policy currently in force.

Contents

Note
In the following list, the required parameters are described first.

GameSession

Object that describes a game session.

Type: GameSession (p. 290) object

Required: No

ProtectionPolicy

Current status of protection for the game session.

- **NoProtection** -- The game session can be terminated during a scale-down event.
- **FullProtection** -- If the game session is in an ACTIVE status, it cannot be terminated during a scale-down event.

Type: String

Valid Values: NoProtection | FullProtection

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2
GameSessionPlacement

Object that describes a StartGameSessionPlacement (p. 204) request. This object includes the full details of the original request plus the current status and start/end time stamps.

Game session placement-related operations include:

- StartGameSessionPlacement (p. 204)
- DescribeGameSessionPlacement (p. 117)
- StopGameSessionPlacement (p. 222)

Contents

**Note**
In the following list, the required parameters are described first.

**EndTime**

Time stamp indicating when this request was completed, canceled, or timed out.

Type: Timestamp

Required: No

**GameProperties**

Set of custom properties for a game session, formatted as key:value pairs. These properties are passed to a game server process in the GameSession (p. 290) object with a request to start a new game session (see Start a Game Session).

Type: Array of GameProperty (p. 289) objects

Array Members: Maximum number of 16 items.

Required: No

**GameSessionArn**

Identifier for the game session created by this placement request. This value is set once the new game session is placed (placement status is FULFILLED). This identifier is unique across all regions. You can use this value as a GameSessionId value as needed.

Type: String


Required: No

**GameSessionData**

Set of custom game session properties, formatted as a single string value. This data is passed to a game server process in the GameSession (p. 290) object with a request to start a new game session (see Start a Game Session).

Type: String


Required: No
GameSessionId

Unique identifier for the game session. This value is set once the new game session is placed (placement status is FULFILLED).

Type: String


Required: No

GameSessionName

Descriptive label that is associated with a game session. Session names do not need to be unique.

Type: String


Required: No

GameSessionQueueName

Descriptive label that is associated with game session queue. Queue names must be unique within each region.

Type: String


Pattern: [a-zA-Z0-9-]+

Required: No

GameSessionRegion

Name of the region where the game session created by this placement request is running. This value is set once the new game session is placed (placement status is FULFILLED).

Type: String


Required: No

IpAddress

IP address of the game session. To connect to a Amazon GameLift game server, an app needs both the IP address and port number. This value is set once the new game session is placed (placement status is FULFILLED).

Type: String

Required: No

MatchmakerData

Information on the matchmaking process for this game. Data is in JSON syntax, formatted as a string. It identifies the matchmaking configuration used to create the match, and contains data on all players assigned to the match, including player attributes and team assignments. For more details on matchmaker data, see Match Data.

Type: String

Required: No

**MaximumPlayerSessionCount**

Maximum number of players that can be connected simultaneously to the game session.

Type: Integer

Valid Range: Minimum value of 0.

Required: No

**PlacedPlayerSessions**

Collection of information on player sessions created in response to the game session placement request. These player sessions are created only once a new game session is successfully placed (placement status is `FULFILLED`). This information includes the player ID (as provided in the placement request) and the corresponding player session ID. Retrieve full player sessions by calling `DescribePlayerSessions` (p. 140) with the player session ID.

Type: Array of `PlacedPlayerSession` (p. 320) objects

Required: No

**PlacementId**

Unique identifier for a game session placement.

Type: String


Pattern: `[a-zA-Z0-9-]+`

Required: No

**PlayerLatencies**

Set of values, expressed in milliseconds, indicating the amount of latency that a player experiences when connected to AWS regions.

Type: Array of `PlayerLatency` (p. 323) objects

Required: No

**Port**

Port number for the game session. To connect to an Amazon GameLift game server, an app needs both the IP address and port number. This value is set once the new game session is placed (placement status is `FULFILLED`).

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 60000.

Required: No

**StartTime**

Time stamp indicating when this request was placed in the queue. Format is a number expressed in Unix time as milliseconds (for example “1469498468.057”).

Type: Timestamp

Required: No
Status

Current status of the game session placement request.
- **PENDING** -- The placement request is currently in the queue waiting to be processed.
- **FULFILLED** -- A new game session and player sessions (if requested) have been successfully created. Values for `GameSessionArn` and `GameSessionRegion` are available.
- **CANCELLED** -- The placement request was canceled with a call to `StopGameSessionPlacement` (p. 222).
- **TIMED_OUT** -- A new game session was not successfully created before the time limit expired. You can resubmit the placement request as needed.

Type: String

Valid Values: PENDING | FULFILLED | CANCELLED | TIMED_OUT

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2
GameSessionQueue

Configuration of a queue that is used to process game session placement requests. The queue configuration identifies several game features:

- The destinations where a new game session can potentially be hosted. Amazon GameLift tries these destinations in an order based on either the queue's default order or player latency information, if provided in a placement request. With latency information, Amazon GameLift can place game sessions where the majority of players are reporting the lowest possible latency.
- The length of time that placement requests can wait in the queue before timing out.
- A set of optional latency policies that protect individual players from high latencies, preventing game sessions from being placed where any individual player is reporting latency higher than a policy's maximum.

Queue-related operations include:

- CreateGameSessionQueue (p. 37)
- DescribeGameSessionQueues (p. 120)
- UpdateGameSessionQueue (p. 250)
- DeleteGameSessionQueue (p. 73)

Contents

**Note**

In the following list, the required parameters are described first.

**Destinations**

List of fleets that can be used to fulfill game session placement requests in the queue. Fleets are identified by either a fleet ARN or a fleet alias ARN. Destinations are listed in default preference order.

Type: Array of GameSessionQueueDestination (p. 303) objects

Required: No

**GameSessionQueueArn**

Amazon Resource Name (ARN) that is assigned to a game session queue and uniquely identifies it. Format is arn:aws:gamelift:<region>::fleet/fleet-a1234567-b8c9-0d1e-2fa3-b45c6d7e8912.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: [a-zA-Z0-9: ]+

Required: No

**Name**

Descriptive label that is associated with game session queue. Queue names must be unique within each region.

Type: String

Pattern: [a-zA-Z0-9-]+

Required: No

**PlayerLatencyPolicies**

Collection of latency policies to apply when processing game sessions placement requests with player latency information. Multiple policies are evaluated in order of the maximum latency value, starting with the lowest latency values. With just one policy, it is enforced at the start of the game session placement for the duration period. With multiple policies, each policy is enforced consecutively for its duration period. For example, a queue might enforce a 60-second policy followed by a 120-second policy, and then no policy for the remainder of the placement.

Type: Array of PlayerLatencyPolicy (p. 324) objects

Required: No

**TimeoutInSeconds**

Maximum time, in seconds, that a new game session placement request remains in the queue. When a request exceeds this time, the game session placement changes to a **TIMED_OUT** status.

Type: Integer

Valid Range: Minimum value of 0.

Required: No

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2
GameSessionQueueDestination

Fleet designated in a game session queue. Requests for new game sessions in the queue are fulfilled by starting a new game session on any destination configured for a queue.

Queue-related operations include:

- CreateGameSessionQueue (p. 37)
- DescribeGameSessionQueues (p. 120)
- UpdateGameSessionQueue (p. 250)
- DeleteGameSessionQueue (p. 73)

Contents

Note

In the following list, the required parameters are described first.

DestinationArn

Amazon Resource Name (ARN) assigned to fleet or fleet alias. ARNs, which include a fleet ID or alias ID and a region name, provide a unique identifier across all regions.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: [a-zA-Z0-9:/-]+

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2
Instance

Properties that describe an instance of a virtual computing resource that hosts one or more game servers. A fleet may contain zero or more instances.

Contents

Note

In the following list, the required parameters are described first.

CreationTime

Time stamp indicating when this data object was created. Format is a number expressed in Unix time as milliseconds (for example "1469498468.057").

Type: Timestamp

Required: No

FleetId

Unique identifier for a fleet that the instance is in.

Type: String

Pattern: ^fleet-\S+

Required: No

InstanceId

Unique identifier for an instance.

Type: String

Pattern: [a-zA-Z0-9\-\.]+

Required: No

IIPAddress

IP address assigned to the instance.

Type: String

Required: No

OperatingSystem

Operating system that is running on this instance.

Type: String

Valid Values: WINDOWS_2012 | AMAZON_LINUX

Required: No

Status

Current status of the instance. Possible statuses include the following:

- PENDING -- The instance is in the process of being created and launching server processes as defined in the fleet's run-time configuration.
• **ACTIVE** -- The instance has been successfully created and at least one server process has successfully launched and reported back to Amazon GameLift that it is ready to host a game session. The instance is now considered ready to host game sessions.

• **TERMINATING** -- The instance is in the process of shutting down. This may happen to reduce capacity during a scaling down event or to recycle resources in the event of a problem.

Type: String

Valid Values: PENDING | ACTIVE | TERMINATING

Required: No

**Type**

EC2 instance type that defines the computing resources of this instance.

Type: String

Valid Values: t2.micro | t2.small | t2.medium | t2.large | c3.large | c3.xlarge | c3.2xlarge | c3.4xlarge | c3.8xlarge | c4.large | c4.xlarge | c4.2xlarge | c4.4xlarge | c4.8xlarge | r3.large | r3.xlarge | r3.2xlarge | r3.4xlarge | r3.8xlarge | r4.large | r4.xlarge | r4.2xlarge | r4.4xlarge | r4.8xlarge | r4.16xlarge | m3.medium | m3.large | m3.xlarge | m3.2xlarge | m3.8xlarge | m4.large | m4.xlarge | m4.2xlarge | m4.4xlarge | m4.10xlarge

Required: No

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2
InstanceAccess

Information required to remotely connect to a fleet instance. Access is requested by calling GetInstanceAccess (p. 160).

Contents

Note
In the following list, the required parameters are described first.

Credentials

Credentials required to access the instance.

Type: InstanceCredentials (p. 308) object

Required: No

FleetId

Unique identifier for a fleet containing the instance being accessed.

Type: String

Pattern: ^fleet-\S+

Required: No

InstanceId

Unique identifier for an instance being accessed.

Type: String

Pattern: [a-zA-Z0-9\-_\.]+

Required: No

IpAddress

IP address assigned to the instance.

Type: String

Required: No

OperatingSystem

Operating system that is running on the instance.

Type: String

Valid Values: WINDOWS_2012 | AMAZON_LINUX

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java
• AWS SDK for Ruby V2
InstanceCredentials

Set of credentials required to remotely access a fleet instance. Access credentials are requested by calling GetInstanceAccess (p. 160) and returned in an InstanceAccess (p. 306) object.

Contents

Note
In the following list, the required parameters are described first.

Secret

Secret string. For Windows instances, the secret is a password for use with Windows Remote Desktop. For Linux instances, it is a private key (which must be saved as a .pem file) for use with SSH.

Type: String

Length Constraints: Minimum length of 1.

Required: No

UserName

User login string.

Type: String

Length Constraints: Minimum length of 1.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2
IpPermission

A range of IP addresses and port settings that allow inbound traffic to connect to server processes on Amazon GameLift. Each game session hosted on a fleet is assigned a unique combination of IP address and port number, which must fall into the fleet's allowed ranges. This combination is included in the GameSession (p. 290) object.

Contents

Note

In the following list, the required parameters are described first.

FromPort

Starting value for a range of allowed port numbers.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 60000.

Required: Yes

IpRange

Range of allowed IP addresses. This value must be expressed in CIDR notation. Example: "0.0.0.0.0.0.0/[subnet mask]" or optionally the shortened version "0.0.0.0/[subnet mask]."

Type: String

Pattern: [^\s]+

Required: Yes

Protocol

Network communication protocol used by the fleet.

Type: String

Valid Values: TCP | UDP

Required: Yes

ToPort

Ending value for a range of allowed port numbers. Port numbers are end-inclusive. This value must be higher than FromPort.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 60000.

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

• AWS SDK for C++
• AWS SDK for Go
• AWS SDK for Java
• AWS SDK for Ruby V2
MatchedPlayerSession

Represents a new player session that is created as a result of a successful FlexMatch match. A successful match automatically creates new player sessions for every player ID in the original matchmaking request.

When players connect to the match’s game session, they must include both player ID and player session ID in order to claim their assigned player slot.

Contents

Note
In the following list, the required parameters are described first.

PlayerId

Unique identifier for a player
Type: String
Required: No

PlayerSessionId

Unique identifier for a player session
Type: String
Pattern: ^psess-\S+
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2
MatchmakingConfiguration

Guidelines for use with FlexMatch to match players into games. All matchmaking requests must specify a matchmaking configuration.

Contents

**Note**
In the following list, the required parameters are described first.

**AcceptanceRequired**
Flag that determines whether or not a match that was created with this configuration must be accepted by the matched players. To require acceptance, set to TRUE.

Type: Boolean
Required: No

**AcceptanceTimeoutSeconds**
Length of time (in seconds) to wait for players to accept a proposed match. If any player rejects the match or fails to accept before the timeout, the ticket continues to look for an acceptable match.

Type: Integer
Required: No

**AdditionalPlayerCount**
Number of player slots in a match to keep open for future players. For example, if the configuration's rule set specifies a match for a single 12-person team, and the additional player count is set to 2, only 10 players are selected for the match.

Type: Integer
Valid Range: Minimum value of 0.
Required: No

**CreationTime**
Time stamp indicating when this data object was created. Format is a number expressed in Unix time as milliseconds (for example "1469498468.057").

Type: Timestamp
Required: No

**CustomEventData**
Information to attached to all events related to the matchmaking configuration.

Type: String
Length Constraints: Minimum length of 0. Maximum length of 256.
Required: No

**Description**
Descriptive label that is associated with matchmaking configuration.
Type: String


Required: No

**GameProperties**

Set of custom properties for a game session, formatted as key:value pairs. These properties are passed to a game server process in the GameSession (p. 290) object with a request to start a new game session (see Start a Game Session). This information is added to the new GameSession (p. 290) object that is created for a successful match.

Type: Array of GameProperty (p. 289) objects

Array Members: Maximum number of 16 items.

Required: No

**GameSessionData**

Set of custom game session properties, formatted as a single string value. This data is passed to a game server process in the GameSession (p. 290) object with a request to start a new game session (see Start a Game Session). This information is added to the new GameSession (p. 290) object that is created for a successful match.

Type: String


Required: No

**GameSessionQueueArns**

Amazon Resource Name (ARN) that is assigned to a game session queue and uniquely identifies it. Format is `arn:aws:gamelift:<region>::fleet/fleet-a1234567-b8c9-0d1e-2fa3-b45c6d7e8912`. These queues are used when placing game sessions for matches that are created with this matchmaking configuration. Queues can be located in any region.

Type: Array of strings

Length Constraints: Minimum length of 1. Maximum length of 256.

Pattern: `[a-zA-Z0-9:/-]+`

Required: No

**Name**

Unique identifier for a matchmaking configuration. This name is used to identify the configuration associated with a matchmaking request or ticket.

Type: String


Pattern: `[a-zA-Z0-9-\./]+`

Required: No

**NotificationTarget**

SNS topic ARN that is set up to receive matchmaking notifications.

Type: String
Length Constraints: Minimum length of 0. Maximum length of 300.
Pattern: [a-zA-Z0-9:_/-]*
Required: No

**RequestTimeoutSeconds**

Maximum duration, in seconds, that a matchmaking ticket can remain in process before timing out. Requests that time out can be resubmitted as needed.

Type: Integer

Required: No

**RuleSetName**

Unique identifier for a matchmaking rule set to use with this configuration. A matchmaking configuration can only use rule sets that are defined in the same region.

Type: String

Pattern: [a-zA-Z0-9-\._]*
Required: No

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2
MatchmakingRuleSet

Set of rule statements, used with FlexMatch, that determine how to build a certain kind of player match. Each rule set describes a type of group to be created and defines the parameters for acceptable player matches. Rule sets are used in MatchmakingConfiguration (p. 312) objects.

A rule set may define the following elements for a match. For detailed information and examples showing how to construct a rule set, see Build a FlexMatch Rule Set.

- Teams -- Required. A rule set must define one or multiple teams for the match and set minimum and maximum team sizes. For example, a rule set might describe a 4x4 match that requires all eight slots to be filled.
- Player attributes -- Optional. These attributes specify a set of player characteristics to evaluate when looking for a match. Matchmaking requests that use a rule set with player attributes must provide the corresponding attribute values. For example, an attribute might specify a player's skill or level.
- Rules -- Optional. Rules define how to evaluate potential players for a match based on player attributes. A rule might specify minimum requirements for individual players, teams, or entire matches. For example, a rule might require each player to meet a certain skill level, each team to have at least one player in a certain role, or the match to have a minimum average skill level. or may describe an entire group--such as all teams must be evenly matched or have at least one player in a certain role.
- Expansions -- Optional. Expansions allow you to relax the rules after a period of time when no acceptable matches are found. This feature lets you balance getting players into games in a reasonable amount of time instead of making them wait indefinitely for the best possible match. For example, you might use an expansion to increase the maximum skill variance between players after 30 seconds.

Contents

Note
In the following list, the required parameters are described first.

RuleSetBody

Collection of matchmaking rules, formatted as a JSON string. (Note that comments are not allowed in JSON, but most elements support a description field.)

Type: String


Required: Yes

CreationTime

Time stamp indicating when this data object was created. Format is a number expressed in Unix time as milliseconds (for example "1469498468.057").

Type: Timestamp

Required: No

RuleSetName

Unique identifier for a matchmaking rule set

Type: String

Pattern: [a-zA-Z0-9-\.]+

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2
MatchmakingTicket

Ticket generated to track the progress of a matchmaking request. Each ticket is uniquely identified by a ticket ID, supplied by the requester, when creating a matchmaking request with StartMatchmaking (p. 214). Tickets can be retrieved by calling DescribeMatchmaking (p. 131) with the ticket ID.

Contents

Note
In the following list, the required parameters are described first.

ConfigurationName

Name of the MatchmakingConfiguration (p. 312) that is used with this ticket. Matchmaking configurations determine how players are grouped into a match and how a new game session is created for the match.

Type: String


Pattern: [a-zA-Z0-9-\._]+

Required: No

EndTime

Time stamp indicating when this matchmaking request stopped being processed due to success, failure, or cancellation. Format is a number expressed in Unix time as milliseconds (for example "1469498468.057").

Type: Timestamp

Required: No

EstimatedWaitTime

Average amount of time (in seconds) that players are currently waiting for a match. If there is not enough recent data, this property may be empty.

Type: Integer

Valid Range: Minimum value of 0.

Required: No

GameSessionConnectionInfo

Identifier and connection information of the game session created for the match. This information is added to the ticket only after the matchmaking request has been successfully completed.

Type: GameSessionConnectionInfo (p. 294) object

Required: No

Players

A set of Player objects, each representing a player to find matches for. Players are identified by a unique player ID and may include latency data for use during matchmaking. If the ticket is in status COMPLETED, the Player objects include the team the players were assigned to in the resulting match.
Type: Array of Player (p. 321) objects

Required: No

**Start Time**

Time stamp indicating when this matchmaking request was received. Format is a number expressed in Unix time as milliseconds (for example "1469498468.057").

Type: Timestamp

Required: No

**Status**

Current status of the matchmaking request.

- **QUEUED** -- The matchmaking request has been received and is currently waiting to be processed.
- **SEARCHING** -- The matchmaking request is currently being processed.
- **REQUIRES_ACCEPTANCE** -- A match has been proposed and the players must accept the match (see AcceptMatch (p. 7)). This status is used only with requests that use a matchmaking configuration with a player acceptance requirement.
- **PLACING** -- The FlexMatch engine has matched players and is in the process of placing a new game session for the match.
- **COMPLETED** -- Players have been matched and a game session is ready to host the players. A ticket in this state contains the necessary connection information for players.
- **FAILED** -- The matchmaking request was not completed. Tickets with players who fail to accept a proposed match are placed in FAILED status.
- **CANCELL ED** -- The matchmaking request was canceled with a call to StopMatchmaking (p. 225).
- **TIMED_OUT** -- The matchmaking request was not successful within the duration specified in the matchmaking configuration.

**Note**

Matchmaking requests that fail to successfully complete (statuses FAILED, CANCELLED, TIMED_OUT) can be resubmitted as new requests with new ticket IDs.

Type: String

Valid Values: CANCELLED | COMPLETED | FAILED | PLACING | QUEUED | REQUIRES_ACCEPTANCE | SEARCHING | TIMED_OUT

Required: No

**Status Message**

Additional information about the current status.

Type: String

Required: No

**Status Reason**

Code to explain the current status. For example, a status reason may indicate when a ticket has returned to SEARCHING status after a proposed match fails to receive player acceptances.

Type: String

Required: No

**TicketId**

Unique identifier for a matchmaking ticket.
Type: String


Pattern: [a-zA-Z0-9-\._]+  

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2
PlacedPlayerSession

Information about a player session that was created as part of a StartGameSessionPlacement (p. 204) request. This object contains only the player ID and player session ID. To retrieve full details on a player session, call DescribePlayerSessions (p. 140) with the player session ID.

Player-session-related operations include:

- CreatePlayerSession (p. 49)
- CreatePlayerSessions (p. 52)
- DescribePlayerSessions (p. 140)
- Game session placements
  - StartGameSessionPlacement (p. 204)
  - DescribeGameSessionPlacement (p. 117)
  - StopGameSessionPlacement (p. 222)

Contents

Note

In the following list, the required parameters are described first.

PlayerId

Unique identifier for a player that is associated with this player session.

Type: String


Required: No

PlayerSessionId

Unique identifier for a player session.

Type: String

Pattern: ^psess-\S+

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2
Player

Represents a player in matchmaking. When starting a matchmaking request, a player has a player ID, attributes, and may have latency data. Team information is added after a match has been successfully completed.

Contents

Note
In the following list, the required parameters are described first.

LatencyInMs

Set of values, expressed in milliseconds, indicating the amount of latency that a player experiences when connected to AWS regions. If this property is present, FlexMatch considers placing the match only in regions for which latency is reported.

If a matchmaker has a rule that evaluates player latency, players must report latency in order to be matched. If no latency is reported in this scenario, FlexMatch assumes that no regions are available to the player and the ticket is not matchable.

Type: String to integer map

Key Length Constraints: Minimum length of 1.

Valid Range: Minimum value of 1.

Required: No

PlayerAttributes

Collection of key:value pairs containing player information for use in matchmaking. Player attribute keys must match the playerAttributes used in a matchmaking rule set. Example: "PlayerAttributes": {"skill": {"N": "23"}, "gameMode": {"S": "deathmatch"}}.

Type: String to AttributeValue (p. 268) object map

Key Length Constraints: Minimum length of 1. Maximum length of 1024.

Required: No

PlayerId

Unique identifier for a player

Type: String


Required: No

Team

Name of the team that the player is assigned to in a match. Team names are defined in a matchmaking rule set.

Type: String


Required: No
See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2
PlayerLatency

Regional latency information for a player, used when requesting a new game session with StartGameSessionPlacement (p. 204). This value indicates the amount of time lag that exists when the player is connected to a fleet in the specified region. The relative difference between a player's latency values for multiple regions are used to determine which fleets are best suited to place a new game session for the player.

Contents

**Note**
In the following list, the required parameters are described first.

**LatencyInMilliseconds**
Amount of time that represents the time lag experienced by the player when connected to the specified region.

Type: Float
Required: No

**PlayerId**
Unique identifier for a player associated with the latency data.

Type: String
Required: No

**RegionIdentifier**
Name of the region that is associated with the latency value.

Type: String
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2
PlayerLatencyPolicy

Queue setting that determines the highest latency allowed for individual players when placing a game session. When a latency policy is in force, a game session cannot be placed at any destination in a region where a player is reporting latency higher than the cap. Latency policies are only enforced when the placement request contains player latency information.

Queue-related operations include:

- CreateGameSessionQueue (p. 37)
- DescribeGameSessionQueues (p. 120)
- UpdateGameSessionQueue (p. 250)
- DeleteGameSessionQueue (p. 73)

Contents

**Note**

In the following list, the required parameters are described first.

**MaximumIndividualPlayerLatencyMilliseconds**

The maximum latency value that is allowed for any player, in milliseconds. All policies must have a value set for this property.

Type: Integer

Valid Range: Minimum value of 0.

Required: No

**PolicyDurationSeconds**

The length of time, in seconds, that the policy is enforced while placing a new game session. A null value for this property means that the policy is enforced until the queue times out.

Type: Integer

Valid Range: Minimum value of 0.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2
PlayerSession

Properties describing a player session. Player session objects are created either by creating a player session for a specific game session, or as part of a game session placement. A player session represents either a player reservation for a game session (status RESERVED) or actual player activity in a game session (status ACTIVE). A player session object (including player data) is automatically passed to a game session when the player connects to the game session and is validated.

When a player disconnects, the player session status changes to COMPLETED. Once the session ends, the player session object is retained for 30 days and then removed.

Player-session-related operations include:

- CreatePlayerSession (p. 49)
- CreatePlayerSessions (p. 52)
- DescribePlayerSessions (p. 140)
- Game session placements
  - StartGameSessionPlacement (p. 204)
  - DescribeGameSessionPlacement (p. 117)
  - StopGameSessionPlacement (p. 222)

Contents

**Note**
In the following list, the required parameters are described first.

**CreationTime**

Time stamp indicating when this data object was created. Format is a number expressed in Unix time as milliseconds (for example "1469498468.057").

Type: Timestamp

Required: No

**FleetId**

Unique identifier for a fleet that the player's game session is running on.

Type: String

Pattern: ^fleet-\S+

Required: No

**GameSessionId**

Unique identifier for the game session that the player session is connected to.

Type: String


Required: No

**IpAddress**

IP address of the game session. To connect to a Amazon GameLift game server, an app needs both the IP address and port number.
Type: String
Required: No

**PlayerData**

Developer-defined information related to a player. Amazon GameLift does not use this data, so it can be formatted as needed for use in the game.

Type: String

Length Constraints: Minimum length of 1. Maximum length of 2048.

Required: No

**PlayerId**

Unique identifier for a player that is associated with this player session.

Type: String


Required: No

**PlayerSessionId**

Unique identifier for a player session.

Type: String

Pattern: ^psess-\S+

Required: No

**Port**

Port number for the game session. To connect to a Amazon GameLift server process, an app needs both the IP address and port number.

Type: Integer

Valid Range: Minimum value of 1. Maximum value of 60000.

Required: No

**Status**

Current status of the player session.

Possible player session statuses include the following:

- **RESERVED** -- The player session request has been received, but the player has not yet connected to the server process and/or been validated.
- **ACTIVE** -- The player has been validated by the server process and is currently connected.
- **COMPLETED** -- The player connection has been dropped.
- **TIMEDOUT** -- A player session request was received, but the player did not connect and/or was not validated within the timeout limit (60 seconds).

Type: String

Valid Values: RESERVED | ACTIVE | COMPLETED | TIMEDOUT

Required: No
TerminationTime

Time stamp indicating when this data object was terminated. Format is a number expressed in Unix time as milliseconds (for example "1469498468.057").

Type: Timestamp
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2
ResourceCreationLimitPolicy

Policy that limits the number of game sessions a player can create on the same fleet. This optional policy gives game owners control over how players can consume available game server resources. A resource creation policy makes the following statement: "An individual player can create a maximum number of new game sessions within a specified time period".

The policy is evaluated when a player tries to create a new game session. For example, with a policy of 10 new game sessions and a time period of 60 minutes, on receiving a CreateGameSession request, Amazon GameLift checks that the player (identified by CreatorId) has created fewer than 10 game sessions in the past 60 minutes.

Contents

Note
In the following list, the required parameters are described first.

NewGameSessionsPerCreator
Maximum number of game sessions that an individual can create during the policy period.
Type: Integer
Valid Range: Minimum value of 0.
Required: No

PolicyPeriodInMinutes
Time span used in evaluating the resource creation limit policy.
Type: Integer
Valid Range: Minimum value of 0.
Required: No

See Also
For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2
**RoutingStrategy**

Routing configuration for a fleet alias.

Fleet-related operations include:

- CreateFleet (p. 19)
- ListFleets (p. 173)
- DeleteFleet (p. 70)
- Describe fleets:
  - DescribeFleetAttributes (p. 93)
  - DescribeFleetCapacity (p. 97)
  - DescribeFleetPortSettings (p. 107)
  - DescribeFleetUtilization (p. 110)
  - DescribeRuntimeConfiguration (p. 144)
  - DescribeEC2InstanceLimits (p. 90)
  - DescribeFleetEvents (p. 101)
- Update fleets:
  - UpdateFleetAttributes (p. 234)
  - UpdateFleetCapacity (p. 238)
  - UpdateFleetPortSettings (p. 242)
  - UpdateRuntimeConfiguration (p. 258)
- Manage fleet actions:
  - StartFleetActions (p. 201)
  - StopFleetActions (p. 219)

**Contents**

**Note**

In the following list, the required parameters are described first.

**FleetId**

Unique identifier for a fleet that the alias points to.

Type: String

Pattern: ^fleet-\S+

Required: No

**Message**

Message text to be used with a terminal routing strategy.

Type: String

Required: No

**Type**

Type of routing strategy.

Possible routing types include the following:
• **SIMPLE** -- The alias resolves to one specific fleet. Use this type when routing to active fleets.

• **TERMINAL** -- The alias does not resolve to a fleet but instead can be used to display a message to the user. A terminal alias throws a TerminalRoutingStrategyException with the RoutingStrategy (p. 329) message embedded.

Type: String

Valid Values: SIMPLE | TERMINAL

Required: No

### See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2
RuntimeConfiguration

A collection of server process configurations that describe what processes to run on each instance in a fleet. All fleets must have a run-time configuration. Each instance in the fleet launches the server processes specified in the run-time configuration and launches new ones as existing processes end. Each instance regularly checks for an updated run-time configuration and follows the new instructions.

The run-time configuration enables the instances in a fleet to run multiple processes simultaneously. Potential scenarios are as follows: (1) Run multiple processes of a single game server executable to maximize usage of your hosting resources. (2) Run one or more processes of different build executables, such as your game server executable and a related program, or two or more different versions of a game server. (3) Run multiple processes of a single game server but with different launch parameters, for example to run one process on each instance in debug mode.

A Amazon GameLift instance is limited to 50 processes running simultaneously. A run-time configuration must specify fewer than this limit. To calculate the total number of processes specified in a run-time configuration, add the values of the ConcurrentExecutions parameter for each ServerProcess (p. 338) object in the run-time configuration.

Fleet-related operations include:

- CreateFleet (p. 19)
- ListFleets (p. 173)
- DeleteFleet (p. 70)
- Describe fleets:
  - DescribeFleetAttributes (p. 93)
  - DescribeFleetCapacity (p. 97)
  - DescribeFleetPortSettings (p. 107)
  - DescribeFleetUtilization (p. 110)
  - DescribeRuntimeConfiguration (p. 144)
  - DescribeEC2InstanceLimits (p. 90)
  - DescribeFleetEvents (p. 101)
- Update fleets:
  - UpdateFleetAttributes (p. 234)
  - UpdateFleetCapacity (p. 238)
  - UpdateFleetPortSettings (p. 242)
  - UpdateRuntimeConfiguration (p. 258)
- Manage fleet actions:
  - StartFleetActions (p. 201)
  - StopFleetActions (p. 219)

Contents

Note
In the following list, the required parameters are described first.

GameSessionActivationTimeoutSeconds

Maximum amount of time (in seconds) that a game session can remain in status ACTIVATING. If the game session is not active before the timeout, activation is terminated and the game session status is changed to TERMINATED.
MaxConcurrentGameSessionActivations

Maximum number of game sessions with status ACTIVATING to allow on an instance simultaneously. This setting limits the amount of instance resources that can be used for new game activations at any one time.

Type: Integer
Required: No

ServerProcesses

Collection of server process configurations that describe which server processes to run on each instance in a fleet.

Type: Array of ServerProcess (p. 338) objects
Array Members: Minimum number of 1 item. Maximum number of 50 items.
Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2
S3Location

Location in Amazon Simple Storage Service (Amazon S3) where build files can be stored for access by Amazon GameLift. This location is specified in a CreateBuild (p. 13) request. For more details, see the Create a Build with Files in Amazon S3.

Contents

Note

In the following list, the required parameters are described first.

Bucket

Amazon S3 bucket identifier. This is the name of your S3 bucket.

Type: String

Length Constraints: Minimum length of 1.

Required: No

Key

Name of the zip file containing your build files.

Type: String

Length Constraints: Minimum length of 1.

Required: No

RoleArn

Amazon Resource Name (ARN) for the access role that allows Amazon GameLift to access your S3 bucket.

Type: String

Length Constraints: Minimum length of 1.

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2
ScalingPolicy

Rule that controls how a fleet is scaled. Scaling policies are uniquely identified by the combination of name and fleet ID.

Operations related to fleet capacity scaling include:

- DescribeFleetCapacity (p. 97)
- UpdateFleetCapacity (p. 238)
- DescribeEC2InstanceLimits (p. 90)
- Manage scaling policies:
  - PutScalingPolicy (p. 178) (auto-scaling)
  - DescribeScalingPolicies (p. 147) (auto-scaling)
  - DeleteScalingPolicy (p. 77) (auto-scaling)
- Manage fleet actions:
  - StartFleetActions (p. 201)
  - StopFleetActions (p. 219)

Contents

**Note**
In the following list, the required parameters are described first.

**ComparisonOperator**

Comparison operator to use when measuring a metric against the threshold value.

Type: String

Valid Values: GreaterThanOrEqualToThreshold | GreaterThanThreshold | LessThanThreshold | LessThanOrEqualToThreshold

Required: No

**EvaluationPeriods**

Length of time (in minutes) the metric must be at or beyond the threshold before a scaling event is triggered.

Type: Integer

Valid Range: Minimum value of 1.

Required: No

**FleetId**

Unique identifier for a fleet that is associated with this scaling policy.

Type: String

Pattern: ^fleet-\S+

Required: No

**MetricName**

Name of the Amazon GameLift-defined metric that is used to trigger a scaling adjustment. For detailed descriptions of fleet metrics, see Monitor Amazon GameLift with Amazon CloudWatch.
• **ActivatingGameSessions** -- Game sessions in the process of being created.
• **ActiveGameSessions** -- Game sessions that are currently running.
• **ActiveInstances** -- Fleet instances that are currently running at least one game session.
• **AvailableGameSessions** -- Additional game sessions that fleet could host simultaneously, given current capacity.
• **AvailablePlayerSessions** -- Empty player slots in currently active game sessions. This includes game sessions that are not currently accepting players. Reserved player slots are not included.
• **CurrentPlayerSessions** -- Player slots in active game sessions that are being used by a player or are reserved for a player.
• **IdleInstances** -- Active instances that are currently hosting zero game sessions.
• **PercentAvailableGameSessions** -- Unused percentage of the total number of game sessions that a fleet could host simultaneously, given current capacity. Use this metric for a target-based scaling policy.
• **PercentIdleInstances** -- Percentage of the total number of active instances that are hosting zero game sessions.
• **QueueDepth** -- Pending game session placement requests, in any queue, where the current fleet is the top-priority destination.
• **WaitTime** -- Current wait time for pending game session placement requests, in any queue, where the current fleet is the top-priority destination.

**Type:** String

**Valid Values:** ActivatingGameSessions | ActiveGameSessions | ActiveInstances | AvailableGameSessions | AvailablePlayerSessions | CurrentPlayerSessions | IdleInstances | PercentAvailableGameSessions | PercentIdleInstances | QueueDepth | WaitTime

**Required:** No

**Name**

Descriptive label that is associated with a scaling policy. Policy names do not need to be unique.

**Type:** String

**Length Constraints:** Minimum length of 1. Maximum length of 1024.

**Required:** No

**PolicyType**

Type of scaling policy to create. For a target-based policy, set the parameter `MetricName` to 'PercentAvailableGameSessions' and specify a `TargetConfiguration`. For a rule-based policy set the following parameters: `MetricName`, `ComparisonOperator`, `Threshold`, `EvaluationPeriods`, `ScalingAdjustmentType`, and `ScalingAdjustment`.

**Type:** String

**Valid Values:** RuleBased | TargetBased

**Required:** No

**ScalingAdjustment**

Amount of adjustment to make, based on the scaling adjustment type.

**Type:** Integer

**Required:** No
ScalingAdjustmentType

Type of adjustment to make to a fleet's instance count (see FleetCapacity (p. 285)):

- **ChangeInCapacity** -- add (or subtract) the scaling adjustment value from the current instance count. Positive values scale up while negative values scale down.
- **ExactCapacity** -- set the instance count to the scaling adjustment value.
- **PercentChangeInCapacity** -- increase or reduce the current instance count by the scaling adjustment, read as a percentage. Positive values scale up while negative values scale down.

Type: String

Valid Values: ChangeInCapacity | ExactCapacity | PercentChangeInCapacity

Required: No

Status

Current status of the scaling policy. The scaling policy can be in force only when in an **ACTIVE** status. Scaling policies can be suspended for individual fleets (see StopFleetActions (p. 219); if suspended for a fleet, the policy status does not change. View a fleet's stopped actions by calling DescribeFleetCapacity (p. 97).

- **ACTIVE** -- The scaling policy can be used for auto-scaling a fleet.
- **UPDATE_REQUESTED** -- A request to update the scaling policy has been received.
- **UPDATING** -- A change is being made to the scaling policy.
- **DELETE_REQUESTED** -- A request to delete the scaling policy has been received.
- **DELETING** -- The scaling policy is being deleted.
- **DELETED** -- The scaling policy has been deleted.
- **ERROR** -- An error occurred in creating the policy. It should be removed and recreated.

Type: String

Valid Values: ACTIVE | UPDATE_REQUESTED | UPDATING | DELETE_REQUESTED | DELETING | DELETED | ERROR

Required: No

TargetConfiguration

Object that contains settings for a target-based scaling policy.

Type: TargetConfiguration (p. 339) object

Required: No

Threshold

Metric value used to trigger a scaling event.

Type: Double

Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
• AWS SDK for Java
• AWS SDK for Ruby V2
ServerProcess

A set of instructions for launching server processes on each instance in a fleet. Each instruction set identifies the location of the server executable, optional launch parameters, and the number of server processes with this configuration to maintain concurrently on the instance. Server process configurations make up a fleet's RuntimeConfiguration (p. 331).

Contents

Note
In the following list, the required parameters are described first.

ConcurrentExecutions

Number of server processes using this configuration to run concurrently on an instance.

Type: Integer

Valid Range: Minimum value of 1.

Required: Yes

LaunchPath

Location of the server executable in a game build. All game builds are installed on instances at the root: for Windows instances C:\game, and for Linux instances /local/game. A Windows game build with an executable file located at MyGame\latest\server.exe must have a launch path of "C:\game\MyGame\latest\server.exe". A Linux game build with an executable file located at MyGame/latest/server.exe must have a launch path of "/local/game/MyGame/latest/server.exe".

Type: String


Required: Yes

Parameters

Optional list of parameters to pass to the server executable on launch.

Type: String


Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2
TargetConfiguration

Settings for a target-based scaling policy (see ScalingPolicy (p. 334). A target-based policy tracks a particular fleet metric specifies a target value for the metric. As player usage changes, the policy triggers Amazon GameLift to adjust capacity so that the metric returns to the target value. The target configuration specifies settings as needed for the target based policy, including the target value.

Operations related to fleet capacity scaling include:

- DescribeFleetCapacity (p. 97)
- UpdateFleetCapacity (p. 238)
- DescribeEC2InstanceLimits (p. 90)

Manage scaling policies:
- PutScalingPolicy (p. 178) (auto-scaling)
- DescribeScalingPolicies (p. 147) (auto-scaling)
- DeleteScalingPolicy (p. 77) (auto-scaling)

Manage fleet actions:
- StartFleetActions (p. 201)
- StopFleetActions (p. 219)

Contents

Note
In the following list, the required parameters are described first.

TargetValue

Desired value to use with a target-based scaling policy. The value must be relevant for whatever metric the scaling policy is using. For example, in a policy using the metric PercentAvailableGameSessions, the target value should be the preferred size of the fleet's buffer (the percent of capacity that should be idle and ready for new game sessions).

Type: Double

Required: Yes

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2
VpcPeeringAuthorization

Represents an authorization for a VPC peering connection between the VPC for an Amazon GameLift fleet and another VPC on an account you have access to. This authorization must exist and be valid for the peering connection to be established. Authorizations are valid for 24 hours after they are issued.

VPC peering connection operations include:

- CreateVpcPeeringAuthorization (p. 56)
- DescribeVpcPeeringAuthorizations (p. 152)
- DeleteVpcPeeringAuthorization (p. 80)
- CreateVpcPeeringConnection (p. 61)
- DescribeVpcPeeringConnections (p. 154)
- DeleteVpcPeeringConnection (p. 82)

Contents

Note
In the following list, the required parameters are described first.

CreationTime

Time stamp indicating when this authorization was issued. Format is a number expressed in Unix time as milliseconds (for example "1469498468.057").

Type: Timestamp

Required: No

ExpirationTime

Time stamp indicating when this authorization expires (24 hours after issuance). Format is a number expressed in Unix time as milliseconds (for example "1469498468.057").

Type: Timestamp

Required: No

GameLiftAwsAccountId

Unique identifier for the AWS account that you use to manage your Amazon GameLift fleet. You can find your Account ID in the AWS Management Console under account settings.

Type: String


Required: No

PeerVpcAwsAccountId

Type: String


Required: No
PeerVpcId

Unique identifier for a VPC with resources to be accessed by your Amazon GameLift fleet. The VPC must be in the same region where your fleet is deployed. To get VPC information, including IDs, use the Virtual Private Cloud service tools, including the VPC Dashboard in the AWS Management Console.

Type: String


Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2
VpcPeeringConnection

Represents a peering connection between a VPC on one of your AWS accounts and the VPC for your Amazon GameLift fleets. This record may be for an active peering connection or a pending connection that has not yet been established.

VPC peering connection operations include:

- CreateVpcPeeringAuthorization (p. 56)
- DescribeVpcPeeringAuthorizations (p. 152)
- DeleteVpcPeeringAuthorization (p. 80)
- CreateVpcPeeringConnection (p. 61)
- DescribeVpcPeeringConnections (p. 154)
- DeleteVpcPeeringConnection (p. 82)

Contents

Note
In the following list, the required parameters are described first.

**FleetId**

Unique identifier for a fleet. This ID determines the ID of the Amazon GameLift VPC for your fleet.

Type: String

Pattern: ^fleet-\S+

Required: No

**GameLiftVpcId**

Unique identifier for the VPC that contains the Amazon GameLift fleet for this connection. This VPC is managed by Amazon GameLift and does not appear in your AWS account.

Type: String


Required: No

**IpV4CidrBlock**

CIDR block of IPv4 addresses assigned to the VPC peering connection for the GameLift VPC. The peered VPC also has an IPv4 CIDR block associated with it; these blocks cannot overlap or the peering connection cannot be created.

Type: String


Required: No

**PeerVpcId**

Unique identifier for a VPC with resources to be accessed by your Amazon GameLift fleet. The VPC must be in the same region where your fleet is deployed. To get VPC information, including IDs, use the Virtual Private Cloud service tools, including the VPC Dashboard in the AWS Management Console.
Type: String


Required: No

**Status**

Object that contains status information about the connection. Status indicates if a connection is pending, successful, or failed.

Type: `VpcPeeringConnectionStatus` (p. 344) object

Required: No

**VpcPeeringConnectionId**

Unique identifier that is automatically assigned to the connection record. This ID is referenced in VPC peering connection events, and is used when deleting a connection with `DeleteVpcPeeringConnection` (p. 82).

Type: String


Required: No

**See Also**

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2
VpcPeeringConnectionStatus

Represents status information for a VPC peering connection. Status is associated with a VpcPeeringConnection (p. 342) object. Status codes and messages are provided from EC2 (see VpcPeeringConnectionStateReason). Connection status information is also communicated as a fleet Event (p. 277).

Contents

Note
In the following list, the required parameters are described first.

Code

Code indicating the status of a VPC peering connection.

Type: String


Required: No

Message

Additional messaging associated with the connection status.

Type: String


Required: No

See Also

For more information about using this API in one of the language-specific AWS SDKs, see the following:

- AWS SDK for C++
- AWS SDK for Go
- AWS SDK for Java
- AWS SDK for Ruby V2
Common Parameters

The following list contains the parameters that all actions use for signing Signature Version 4 requests with a query string. Any action-specific parameters are listed in the topic for that action. For more information about Signature Version 4, see Signature Version 4 Signing Process in the Amazon Web Services General Reference.

**Action**

The action to be performed.

Type: string

Required: Yes

**Version**

The API version that the request is written for, expressed in the format YYYY-MM-DD.

Type: string

Required: Yes

**X-Amz-Algorithm**

The hash algorithm that you used to create the request signature.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Valid Values: AWS4-HMAC-SHA256

Required: Conditional

**X-Amz-Credential**

The credential scope value, which is a string that includes your access key, the date, the region you are targeting, the service you are requesting, and a termination string (“aws4_request”). The value is expressed in the following format: access_key/YYYYMMDD/region/service/aws4_request.

For more information, see Task 2: Create a String to Sign for Signature Version 4 in the Amazon Web Services General Reference.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string

Required: Conditional

**X-Amz-Date**

The date that is used to create the signature. The format must be ISO 8601 basic format (YYYYMMDD'T'HHMMSS'Z'). For example, the following date time is a valid X-Amz-Date value: 20120325T120000Z.

Condition: X-Amz-Date is optional for all requests; it can be used to override the date used for signing requests. If the Date header is specified in the ISO 8601 basic format, X-Amz-Date is
not required. When X-Amz-Date is used, it always overrides the value of the Date header. For more information, see Handling Dates in Signature Version 4 in the Amazon Web Services General Reference.

Type: string
Required: Conditional

X-Amz-Security-Token

The temporary security token that was obtained through a call to AWS Security Token Service (AWS STS). For a list of services that support temporary security credentials from AWS Security Token Service, go to AWS Services That Work with IAM in the IAM User Guide.

Condition: If you're using temporary security credentials from the AWS Security Token Service, you must include the security token.

Type: string
Required: Conditional

X-Amz-Signature

Specifies the hex-encoded signature that was calculated from the string to sign and the derived signing key.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string
Required: Conditional

X-Amz-SignedHeaders

Specifies all the HTTP headers that were included as part of the canonical request. For more information about specifying signed headers, see Task 1: Create a Canonical Request For Signature Version 4 in the Amazon Web Services General Reference.

Condition: Specify this parameter when you include authentication information in a query string instead of in the HTTP authorization header.

Type: string
Required: Conditional
Common Errors

This section lists the errors common to the API actions of all AWS services. For errors specific to an API action for this service, see the topic for that API action.

**AccessDeniedException**

You do not have sufficient access to perform this action.

HTTP Status Code: 400

**IncompleteSignature**

The request signature does not conform to AWS standards.

HTTP Status Code: 400

**InternalFailure**

The request processing has failed because of an unknown error, exception or failure.

HTTP Status Code: 500

**InvalidAction**

The action or operation requested is invalid. Verify that the action is typed correctly.

HTTP Status Code: 400

**InvalidClientTokenId**

The X.509 certificate or AWS access key ID provided does not exist in our records.

HTTP Status Code: 403

**InvalidParameterCombination**

Parameters that must not be used together were used together.

HTTP Status Code: 400

**InvalidParameterValue**

An invalid or out-of-range value was supplied for the input parameter.

HTTP Status Code: 400

**InvalidQueryParameter**

The AWS query string is malformed or does not adhere to AWS standards.

HTTP Status Code: 400

**MalformedQueryString**

The query string contains a syntax error.

HTTP Status Code: 404

**MissingAction**

The request is missing an action or a required parameter.

HTTP Status Code: 400
MissingAuthenticationToken

The request must contain either a valid (registered) AWS access key ID or X.509 certificate.

HTTP Status Code: 403

MissingParameter

A required parameter for the specified action is not supplied.

HTTP Status Code: 400

OptInRequired

The AWS access key ID needs a subscription for the service.

HTTP Status Code: 403

RequestExpired

The request reached the service more than 15 minutes after the date stamp on the request or more than 15 minutes after the request expiration date (such as for pre-signed URLs), or the date stamp on the request is more than 15 minutes in the future.

HTTP Status Code: 400

ServiceUnavailable

The request has failed due to a temporary failure of the server.

HTTP Status Code: 503

ThrottlingException

The request was denied due to request throttling.

HTTP Status Code: 400

ValidationError

The input fails to satisfy the constraints specified by an AWS service.

HTTP Status Code: 400