

# **Game Server Elastic-scaling**

## **Getting Started**

### **Product Documentation**



## Copyright Notice

©2013-2019 Tencent Cloud. All rights reserved.

Copyright in this document is exclusively owned by Tencent Cloud. You must not reproduce, modify, copy or distribute in any way, in whole or in part, the contents of this document without Tencent Cloud's the prior written consent.

## Trademark Notice



All trademarks associated with Tencent Cloud and its services are owned by Tencent Cloud Computing (Beijing) Company Limited and its affiliated companies. Trademarks of third parties referred to in this document are owned by their respective proprietors.

## Service Statement

This document is intended to provide users with general information about Tencent Cloud's products and services only and does not form part of Tencent Cloud's terms and conditions. Tencent Cloud's products or services are subject to change. Specific products and services and the standards applicable to them are exclusively provided for in Tencent Cloud's applicable terms and conditions.

# Contents

## Getting Started

Demo

Auto Scaling

Zero Downtime Updates

Nearby Resource Scheduling

Cross-Region Disaster Recovery

# Getting Started

## Demo

Last updated : 2022-05-06 10:24:04

### Overview

This document describes how to use the "Demo" to get started with GSE, a game server hosting service.

### Directions

#### Step 1: uploading demo package

1. Log in to the [GSE Console](#) and click **Demo** in the left sidebar.
2. Select the service region in the top-left corner and click **Quick Upload of Demo Package**. After the message indicating that the package has been uploaded successfully is displayed, click **Next**.

Getting Started Examples Beijing

1 **Create Package** > 2 Create Fleet > 3 Create Game Server Session

① We have prepared a sample package that has integrated ServerSDK for you. If you want to build it by yourself, you need to prepare ServerSDK. [Package Creation Guide](#) [ServerSDK Integration](#)

Quick Upload of Demo Package

✔ Created successfully

Uploaded demo package successfully. asset-5omva57g

Next

Note :

- The demo package provided by GSE has already integrated the gRPC framework through which the game process communicates with GSE.
- If you want to create on your own, please see [Creating Code Packages](#).

## Step 2: creating a server fleet

Click **Quick Creation of Server Fleet**, and you will see how the creation status changes, such as “Create (In progress)” and “Downloading (In progress)”. When “Created successfully” is displayed, click **Next**.

- To create a server fleet needs a “Completed” status for all 6 steps, namely, Create (Completed), Downloading (Completed), Verifying (Completed), Generating (Completed), Activating (Completed), and Active (Completed).

- **Status:** creating, or server fleet XXX created successfully.

The screenshot displays a three-step wizard for creating a game server session. Step 1, 'Create Package', is completed. Step 2, 'Create Fleet', is the current active step, indicated by a blue circle with the number '2'. Step 3, 'Create Game Server Session', is the next step. A blue information banner at the top of the step 2 area states: 'This step publishes the uploaded package to the server fleet. During initialization, only one server instance runs the [Guide](#)'. Below this banner, a progress bar for 'Quick Creation of Server Fleet' shows 17% completion. To the right of the progress bar, a text box contains the message: 'It may take about 2-3 minutes to create a server fleet in 6 steps. Please wait. Status: creating. Creation time: 2020-08-11 10:12:41' followed by 'Create(In progress)'. At the bottom of the wizard, there are 'Prev' and 'Next' navigation buttons.

✓ Create Package > 
 **2 Create Fleet** > 
 3 Create Game Server Session

ⓘ This step publishes the uploaded package to the server fleet. During initialization, only one server instance runs the pack  
[Guide](#)

Quick Creation of Server Fleet

33 %

It may take about 2-3 minutes to create a server fleet in 6 steps. Please wait.  
 Status: creating. Creation time: 2020-08-11 10:12:41

Create(Completed)

Downloading(In progress)

Prev Next

✓ Create Package > 
 **2 Create Fleet** > 
 3 Create Game Server Session

ⓘ This step publishes the uploaded package to the server fleet. During initialization, only one server instance runs the pack  
[Guide](#)

Quick Creation of Server Fleet

✓ Created successfully

It may take about 2-3 minutes to create a server fleet in 6 steps. Please wait.  
 Status: server fleet fleet-qp3g3fn6-o327b0d8 created successfully

Create(Completed)

Downloading(Completed)

Verifying(Completed)

Generating(Completed)

Prev Next

Note :

- Your demo package is deployed onto the server fleet as it is being created.
- A server fleet consists of a group of servers capable of auto-scaling, so the demo package can be deployed globally with ease.
- If you want to create on your own, please see [Creating Server Fleets](#).

### Step 3: creating a game server session and a player session

- Click **Create Game Server Session**, and a message will be displayed indicating that a game server session has been created successfully.

✓ Create Package > ✓ Create Fleet > 3 Create Game Server Session

To quickly create game server sessions and player sessions, you usually need to call cloud APIs to do so.

Create and select a game server session

Create Game Server Session

NAME\_GSS15971123045

Create and select a player session

Create Player Session

Please select an option

Redirect to client webpage

Game server session NAME\_GSS1597112304522

Prev Complete Click "Complete" to enter the next round of trial. This operation will not delete the created resources

Note :

- This operation calls the `CreateGameServerSession` TencentCloud API so that GSE will create a game server session and assign it a service process.
- If you want to create on your own, please see the API document [CreateGameServerSession](#).
- Click **Create Player Session**, and a message indicating that a player session has been created successfully will be displayed.

✓ Create Package > ✓ Create Fleet > 3 Create Game Server Session

**i** To quickly create game server sessions and player sessions, you usually need to call cloud APIs to do so.

Create and select a game server session

Create Game Server Session

NAME\_GSS15971123045 ▾

Create and select a player session

Create Player Session

psess-1aeb7vay-hu29ihp ▾

Redirect to client webpage

```
Player session psess-1aeb7vay-hu29ihpu:120.53.27.163:59066. Creation time: 2020-08-11 10:18:57. The player (client) needs to connect to the server within 1 minute; otherwise, the connection will expire.
```

Prev Complete Click "Complete" to enter the next round of trial. This operation will not delete the created resource

Note :

- This operation calls the `JoinGameServerSession` TencentCloud API, so that GSE will create a player session and add the player to a game server session.
- If you want to create on your own, please see the API document [JoinGameServerSession](#).

#### Step 4: connecting client to the game server

Click **Redirect to client webpage** to access the page for connecting the client to the game server. Click **Connect**, and a message indicating that the server has been connected successfully will be displayed.

Note :

- After creating the player session, the player (client) needs to connect to the server within 1 minute; otherwise, the connection will expire.
- This demo package is a chat service. When multiple players connect to the server, they can chat with each other.

The above four steps simulate the entire GSE integration process. For more information, please see [Development Guide](#).

# Auto Scaling

Last updated : 2020-08-04 11:30:42

## Overview

This document describes how to implement auto scaling through a server fleet.

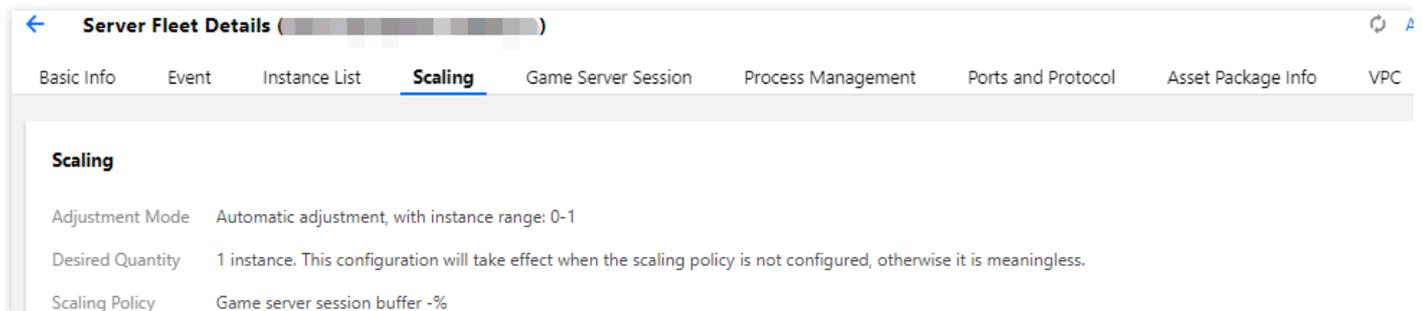
## Prerequisites

You have completed the steps in the [Demo](#).

## Directions

### Modifying the scaling configuration and the number of processes

1. Log in to the [GSE Console](#) and click **Fleet** on the left sidebar.
2. Click the ID of the server fleet created in the "Demo" to enter the fleet details page. Click the **Scaling** tab to view scaling details.



3. Click **Modify** in the top-right corner to modify the scaling configuration as follows:
  - i. Select "Automatic adjustment" as the adjustment mode.
  - ii. Set "Instance Range" to 0-3 so that there will be room for expansion.
  - iii. Set the game server session buffer to 30%, i.e., when the number of game battles (sessions) loaded on the server exceeds 70% of the limit, expansion will be performed.

iv. After making the modifications, click **OK**.

**Modify Scaling Policy**
✕

ID

Name

Instance Type S5.LARGE8

Adjustment Mode Automatic adjustment ▼

Instance Range \* 0 ✔ to 3 ✔

Automatically adjust within the set instance range and do not exceed the range.  
The sum of max instance range in a single region of an account cannot exceed the highest quota of each region. [View Region Quota](#) ↗

Desired Quantity \* - 1 + pcs ✔

Generally, this metric does not take effect in automatic adjustment. However, if the game server session buffer and other automatic adjustment policies are not configured, GSE Service will adjust the number of instances with this metric.

Game Server Session Buffer \* 30 %

This metric represents the proportion of reserved idle instance resources for game server sessions, and automatic adjustment will scale with it.

Scaling Cooldown Time - 10 + min

This metric indicates the time interval between two scaling operations.It can be set from 3 to 30 minutes, which is determined by the duration of the server's process launch.

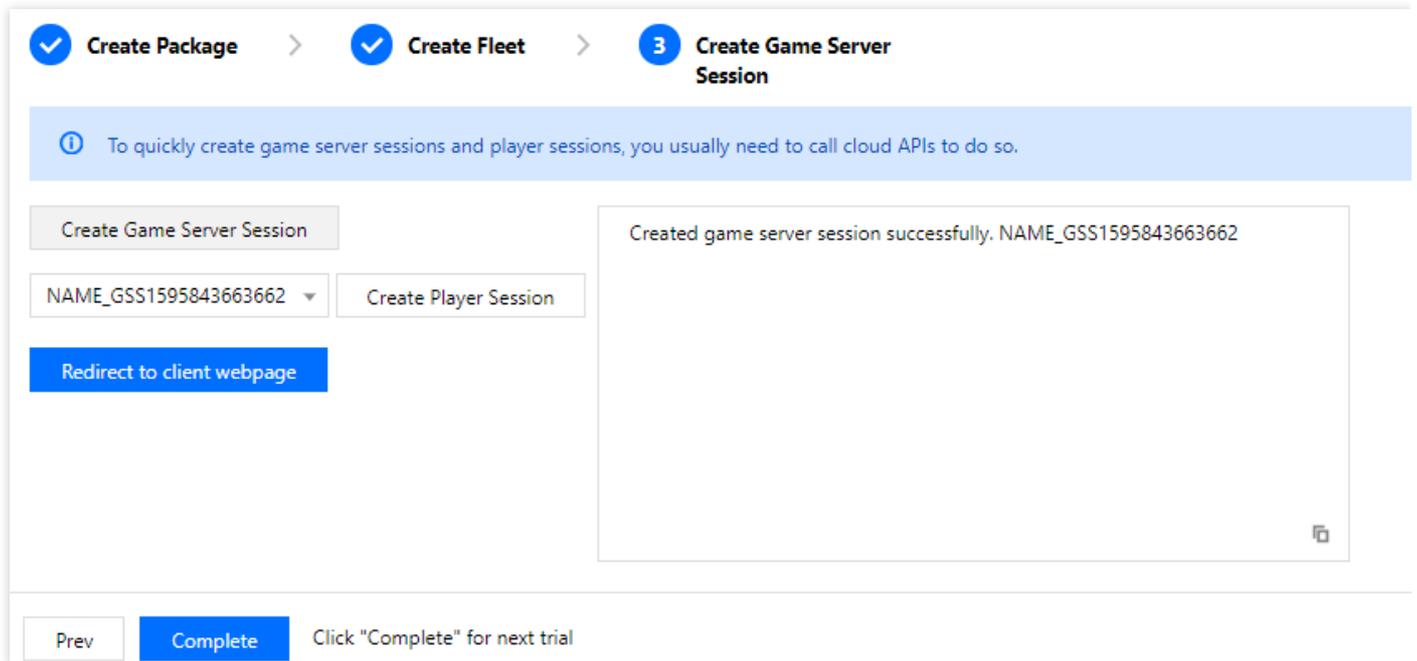
OK
Cancel

### i **Note :**

- $\text{Game server session buffer} = \frac{\text{number of available game server sessions}}{\text{maximum number of game server sessions}}$   
 $= \frac{(\text{maximum number of game server sessions} - \text{number of active game server sessions})}{\text{maximum number of game server sessions}}$ .
- If the game server session buffer is configured as 30%, expansion will be performed when the available game server sessions are below 30%; otherwise, reduction will be performed.

## Creating game server session and observing expansion result

1. In the console, click **Demo** on the left sidebar, complete the first three steps in the [Demo](#), and click **Create Game Server Session** for seven times to create eight game server sessions and trigger expansion.



The screenshot displays a three-step demo workflow in the Tencent Cloud console. The steps are: 1. Create Package (completed), 2. Create Fleet (completed), and 3. Create Game Server Session (current step). A blue information banner at the top states: "To quickly create game server sessions and player sessions, you usually need to call cloud APIs to do so." Below this, there are buttons for "Create Game Server Session" and "Create Player Session". A dropdown menu shows the session ID "NAME\_GSS1595843663662". A blue button labeled "Redirect to client webpage" is also present. On the right, a white box displays the message: "Created game server session successfully. NAME\_GSS1595843663662". At the bottom, there are "Prev" and "Complete" buttons, with a note: "Click 'Complete' for next trial".

### **Note :**

- In GSE Console > Demo, one server can sustain up to 10 game server sessions by default. Therefore, when the server sustains seven game server sessions, the game server session buffer will be number of available game server sessions / maximum number of game server sessions = (maximum number of game server sessions - number of active game server sessions) / maximum number of game server sessions =  $(10 - 7) / 10 = 30\%$ .
- Therefore, you need to create eight game server sessions at least to trigger expansion.

2. Click **Fleet** on the left sidebar, and select the ID of the created server fleet to enter the fleet details page. Now, click the **Instance List** tab and observe. After two minutes, you will see that

the number of instances is increased to two.

Instance ID	Instance Status	Process Count	Game Server Session	Player Sessions	Run Time	Creation Time
ins-6t10upol	Running	0/0	0/0	0/0	0d 0h 2m 1s	2020-07-27 17:55:00
ins-6y6bnt1n	Running	10/10	8/8	0/80	0d 0h 10m 34s	2020-07-27 17:46:27

### Note :

After the creation, **do not** click **Complete** for next round of trial. Instead, you will still need the above configuration for subsequent reduction steps.

## Ending game server session and observing reduction result

- In the console, click **Demo** on the left sidebar and proceed with the subsequent steps after the above expansion. After selecting each game server session, click **Create Player Session** once to create a player session.

- Click **Redirect to client webpage** to enter the client page. Click **Connect** to successfully connect to the server. Click **End Game Session** to end the game server session. Repeat steps 1 and 2 once to end 2 game server sessions and trigger reduction.

**Note :**

- Game server session buffer = number of available game server sessions / maximum number of game server sessions = (maximum number of game server sessions - number of active game server sessions) / maximum number of game server sessions =  $(20 - 6) / 20 = 70\%$ .
- As there are only six active game server sessions left now, making the buffer increased to 70%, reduction is triggered (above 30%).
- Currently, if you close your client webpage, the previously created player sessions will not be able to reconnect to your client. In this case, you have to create a new player session for reconnection in order to close the game server session.

3. Click **Fleet** on the left sidebar, select the ID of the created server fleet to access the fleet details page, and select **Instance List**. Observe the instance quantity. After two minutes, you will see the number of servers decreases to two.

Instance ID	Instance Status	Process Count	Game Server Session	Player Sessions	Run Time	Creation Time
[Redacted]	Running	10/10	0/0	0/0	0d 0h 5m 32s	2020-07-27 17:55:00

# Zero Downtime Updates

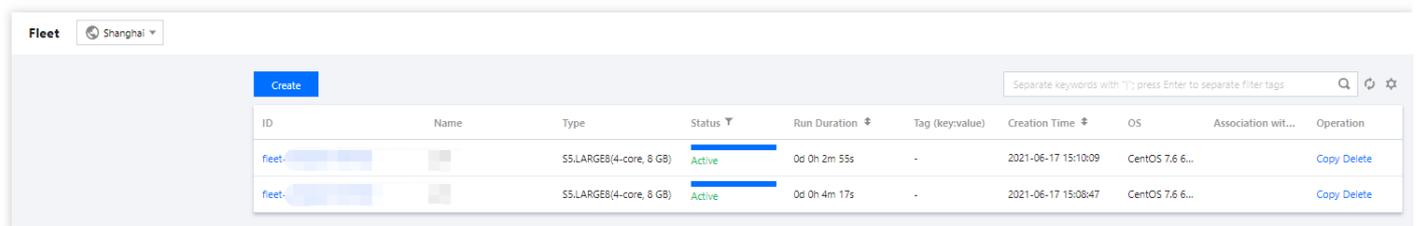
Last updated : 2021-06-28 10:09:36

## Overview

This document describes how to implement zero downtime update through an alias.

## Prerequisites

- Create two server fleets in Shanghai region as instructed below:
  - Complete the first three steps in [Demo](#): click **Quick Upload of Demo Package, Quick Creation of Server Fleet**, and **Create Game Server Session** and then click **Complete**.
- You have created server fleet 1 and server fleet 2 (Shanghai region).



ID	Name	Type	Status	Run Duration	Tag (key:value)	Creation Time	OS	Association wit...	Operation
fleet-...		SS.LARGE8(4-core, 8 GB)	Active	0d 0h 2m 55s	-	2021-06-17 15:10:09	CentOS 7.6 6...		Copy Delete
fleet-...		SS.LARGE8(4-core, 8 GB)	Active	0d 0h 4m 17s	-	2021-06-17 15:08:47	CentOS 7.6 6...		Copy Delete

## Directions

### Creating an alias

1. Log in to the [GSE console](#) and click **Alias** on the left sidebar.
2. Select the service region in the top-left corner and click **Create**.
3. In the alias creation page, enter the name and description, select the alias type and associated server fleet in the corresponding drop-down list, and click **Confirm**.
  - Name: enter the name of the alias for easy identification in the directory, which is "zero downtime update test" in this example.
  - Type: select common alias or terminal alias, which is "common alias" in this example.
    - Common alias: it points to a fleet, under which the system automatically finds servers and assigns them to clients. If you select common alias, you need to associate an available server fleet.

- Terminal alias: it doesn't point to a fleet. You can describe the reason why the alias cannot be used in **Termination Info**, which will be sent to clients.
  - Associate Server Fleet: after selecting common alias, select "server fleet 1".
  - Description: enter a short description of the alias for easy identification, which is "test" in this example.
  - Tag (optional): the tag is used to manage resources by category from different dimensions. If the existing tag does not meet your requirements, please go to the [Tag](#) console to create new tags.
4. After configuring the settings, click **Confirm** to create the alias.

**Create Alias** [Close]

Name \*  ✓

Type \*  ▼

Associate Server Fleet \*  ▼

Description \*

Tag (optional) ⓘ  ▼  ▼ ×

+ Add

## Creating a game server session

Call the TencentCloud API in the code. This example uses [TencentCloud API Explorer](#) for quick creation.

Note :

Input parameter description:

- `Region` : indicates the region, which is "ap-shanghai" (East China (Shanghai)) in this example.

- `MaximumPlayerSessionCount` : indicates the maximum number of players, which is 10 in this example.
- `AliasId` : indicates the alias ID, which is the ID of the newly created alias in this example.

The screenshot shows the 'CreateGameServerSession' API configuration in the Tencent Cloud API Explorer. The 'Input Parameters' section is highlighted with a red box, showing 'Region' set to 'ap-shanghai' and 'MaximumPlayerSessionCount' set to '10'. The 'Send Request' button is also highlighted with a red arrow. The 'Response' section shows a JSON object representing the created game server session.

```

{
  "Response": {
    "GameServerSession": {
      "AvailabilityStatus": "Enable",
      "ConfigurationName": "",
      "CreationTime": "2021-06-17T07:18:25Z",
      "CreatorId": "",
      "CurrentCustomCount": 0,
      "CurrentPlayerSessionCount": 0,
      "DnsName": "",
      "FleetId": "f-...",
      "GameProperties": [],
      "GameServerSessionData": "",
      "GameServerSessionId": "qos:gsse:ap-shanghai:uir-...:gameserver",
      "InstanceId": "i-...",
      "InstanceType": "S5.LARGE8",
      "IpAddress": "1...",
      "MatchmakerData": "",
      "MaxCustomCount": 0,
      "MaximumPlayerSessionCount": 10,
      "Name": "",
      "PlayerSessionCreationPolicy": "ACCEPT_ALL",
      "Port": 39886,
      "ProcessUUID": "i-...",
      "Status": "ACTIVATING",
      "StatusReason": "",
      "TerminationTime": null,
      "Weight": 5
    }
  }
}
    
```

If a game server session is successfully created through [TencentCloud API Explorer](#), you can see that it is generated in server fleet 1.

The screenshot shows the 'Server Fleet Details' page for a specific fleet. The 'Game Server Session' tab is selected, displaying a table with the following data:

Game Server Session ID	Name	Status	Instance Type	IP	Port	Player S...	Creation Time	Run Duration
...	-	Active	S5.LARGE8	...	39886	0	2021-06-17 15:18:25	0d 0h 3m 20s

## Modifying the alias configuration

1. In the console, click **Alias** on the left sidebar to enter the alias list page.
2. Select the created alias and click **Modify** to enter the alias editing page, modify the alias configuration, and set "Associate Server Fleet" to "Server fleet 2".

### Modify Alias

Name \*

Type \*

Associate Server Fleet \*

Description \*

Tag (optional) [+ Add](#)

## Creating another game server session

Note :  
Create another game server session with the same alias.

Create another game server session through [TencentCloud API Explorer](#), and you can see that a game server session is generated in, that is, assigned to server fleet 2.

Server Fleet Details ( )

Basic Info Event Instance List Scaling **Game Server Session** Process Management Ports and Protocol Resource Info VPC

Game Server Session ID	Name	Status	Instance Type	IP	Port	Player S...	Creation Time	Run Duration
gc	-	Active	S5.LARGE8		36262	0	2021-06-17 15:25:35	0d 0h 0m 2s

[Homepage](#) [Next](#)

## Notes on Zero Downtime Updates

For version updates, you can create a new server fleet and point the alias to it.

- Automatic reduction will be performed on the old server fleet as the game server sessions decrease.
- Automatic expansion will be performed on the new server fleet as the game server sessions increase, thus implementing zero downtime update.

# Nearby Resource Scheduling

Last updated : 2021-05-20 10:18:11

## Overview

This document describes how to implement nearby resource scheduling through a game server queue.

## Prerequisites

- Create two server fleets in Shanghai and Silicon Valley regions as instructed below:
  - Complete the first three steps in the [Demo](#): click **Quick Upload of Demo Package**, **Quick Creation of Server Fleet**, and **Create Game Server Session** and then click **Complete**.
- You have created server fleet 1 (Shanghai region).

ID	Name	Type	Status	Run Duration	Creation Time	OS	Operation
fleet-qp3g3caa-fgdv053w	GseDemoFleet	-	Active	4d 20h 48m 34s	2020-06-28 18:29:36	CentOS7.16	Delete

- You have created server fleet 2 (US region).

ID	Name	Type	Status	Run Duration	Creation Time	OS	Operation
fleet-qp3g3ffn-44fgiakh	GseDemoFleet	Standard S3(4-core, 8 ...	Active	0d 0h 3m 4s	2020-07-03 15:20:57	CentOS7.16	Delete

## Directions

### Creating game server queue

1. Log in to the [GSE Console](#) and click **Queue** on the left sidebar to enter the game server queue page.

2. Select the service region in the top-left corner and click **Create**.
3. In the game server queue creation page, enter the basic information:
  - Identifier: enter a valid identifier, which can contain letters only and is "dispatchingnearby" in this example.
  - Timeout Allocation: enter the max time that a game server session request can be retained in a multi-region deployment. It can be up to 600 seconds and is 30 seconds in this example.
4. Enter the latency policies:
  - In the first 10s, server fleets whose latency for any players is up to 80 ms are matched and waited for first.
  - In the subsequent 10s (i.e., in the first 20 seconds), server fleets whose latency for any players is up to 100 ms are matched and waited for first.
  - In the last 10s (= 30s - 10s - 10s) of the timeout period, server fleets whose latency for any players is up to 150 ms are matched and waited for.
5. Select the created server fleet 1 (Shanghai region) and server fleet 2 (US region) as the target.
6. Click **OK** to complete creating the game server queue.

**Basic Info**

Identifier ⓘ \*  ✓

Timeout Allocation ⓘ  sec ✓

**Latency Policy**

Priority	Time Consumed (s) ⓘ	Max Player Delay (ms) ⓘ	Operation
1	<input type="text" value="10"/> sec	<input type="text" value="80"/> ms	<a href="#">Delete</a>
2	<input type="text" value="10"/> sec	<input type="text" value="100"/> ms	<a href="#">Delete</a>
3	Remaining Timeout	<input type="text" value="150"/> ms	

[+ Add Latency Policy](#)

**Target**

Priority	Region	Type	ID and Name	Ope...
1 ↑ ↓	<input type="text" value="Shanghai"/>	<input type="text" value="Fleet"/>	<input type="text" value="fleet-c..."/>	<a href="#">Delete</a>
2 ↑ ↓	<input type="text" value="Silicon Valley"/>	<input type="text" value="Fleet"/>	<input type="text" value="fleet-c..."/>	<a href="#">Delete</a>

[+ Add Target](#)

## Starting placing game server session with queue

Call the `StartGameServerSessionPlacement` TencentCloud API in the code to place the game server session in the server fleet process. This example uses [TencentCloud API Explorer](#) for quick creation.

说明：

#### Input parameter description

- `Region` indicates the region, which is "ap-shanghai" (East China (Shanghai)) in this example;
- `PlacementId` indicates the unique ID of the game server session placement, which is 1 in this example;
- `GameServerSessionQueueName` indicates the game server session queue name, which is "dispatchingnearby" in this example;
- `MaximumPlayerSessionCount` indicates the maximum number of concurrent players allowed by the game server to connect to the game session, which is 2 in this example;
- `DesiredPlayerSessions.N` indicates the player game session information, where `PlayerId` is the unique player ID associated with the player session. In this example, two values of 1 and 2 are entered respectively;
- `PlayerLatencies.N` indicates the player latency, where `PlayerId` is the player ID, `RegionIdentifier` is the name of the region where the latency occurs, and `LatencyInMilliseconds` is the latency in milliseconds. In this example, four value sets are entered, i.e., [1, ap-shanghai, 100], [1, na-siliconvalley, 50], [2, ap-shanghai, 60], and [2, na-siliconvalley, 80].



APIs ✕ 🔍 **StartGameServerSess...** 2019-11-

**Service Management APIs**

- CreateGameServerSession
- DescribeGameServerSessionData
- DescribeGameServerSessionPlace
- DescribeGameServerSessions
- DescribePlayerSessions
- GetGameServerSessionLogUrl
- GetInstanceAccess
- JoinGameServerSession
- SearchGameServerSessions
- StartGameServerSessionPlacem**
- StopGameServerSessionPlacem
- UpdateGameServerSession

**Input Parameters**

View Only Required Parameters

Region

PlacementId

GameServerSessionQueueName

MaximumPlayerSessionCount

DesiredPlayerSessions.N  (Optional)

1

PlayerId

PlayerData

[Add](#)

GameProperties.N  (Optional)

1

Key

Value

[Add](#)

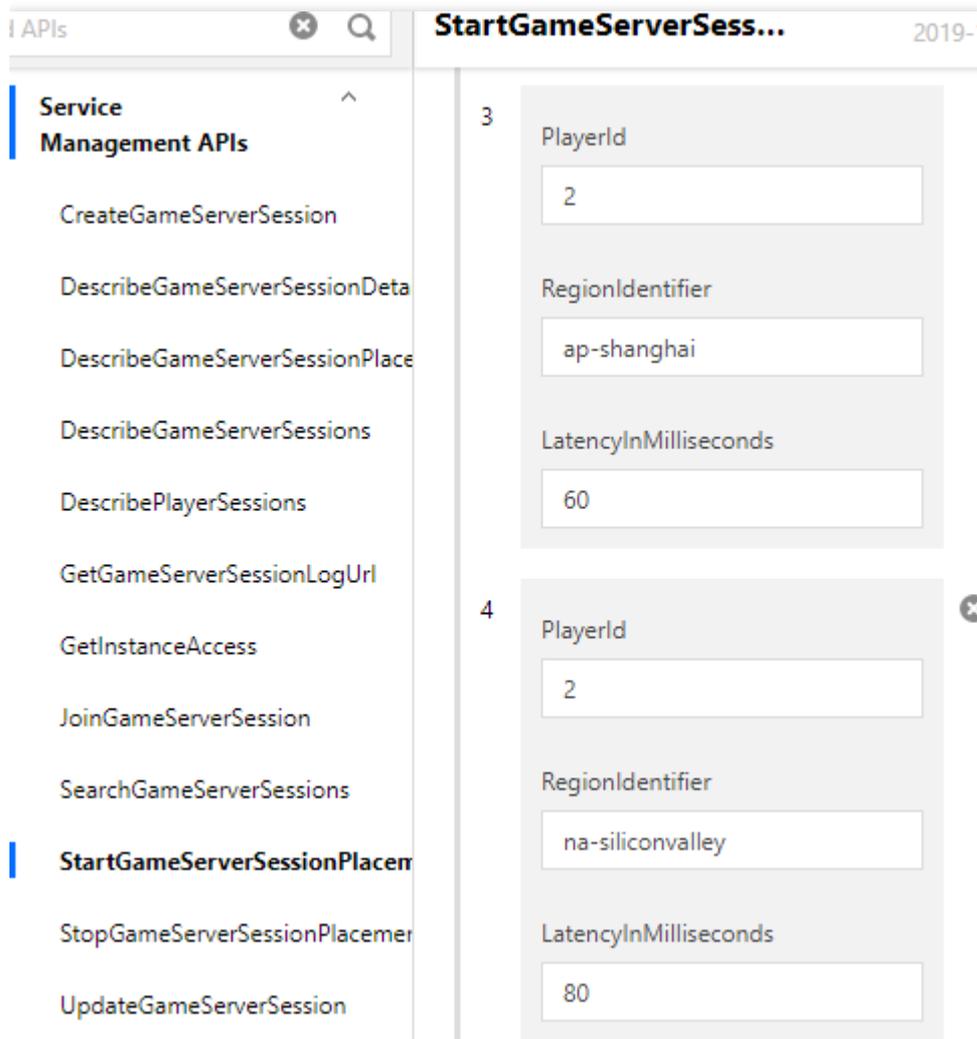
APIs ✕ 🔍 **StartGameServerSess...** 2019-11-

**Service Management APIs**

PlayerLatencies.N  (Optional)

1

CreateGameServerSession	
DescribeGameServerSessionData	
DescribeGameServerSessionPlace	
DescribeGameServerSessions	
DescribePlayerSessions	
GetGameServerSessionLogUrl	
GetInstanceAccess	
JoinGameServerSession	
SearchGameServerSessions	
<b>StartGameServerSessionPlacem</b>	1 PlayerId 1 RegionIdentifier ap-shanghai LatencyInMilliseconds 100
StopGameServerSessionPlacem	2 PlayerId 1 RegionIdentifier na-siliconvalley LatencyInMilliseconds 50
UpdateGameServerSession	



### Scheduling result evaluation of latency policy

Latency in two players' arrival at the target address:

- The latency for player 1 is 100 ms to Shanghai and 50 ms to Silicon Valley.
- The latency for player 2 is 60 ms to Shanghai and 80 ms to Silicon Valley.

Since the latency policy specifies that in the first 10 seconds, servers in regions where the latency for any players is up to 80 ms will be matched first, the game server session will be scheduled to the Silicon Valley region.

### Test result returned after API call:

The game server session is scheduled to server fleet 2 (US region).

← **Server Fleet Details ( fleet- )** [Alarming Configuration](#) [View Monit](#)

Basic Info	Event	Instance List	Scaling	<b>Game Server Session</b>	Process Management	Ports and Protocol	Asset Package Info	VPC
Game Server Sessi...	Name	Status ▼	Instance Type	IP	Port	Player Session	Creation Time ↕	Run Durator
qcs:gse:na-si...	-	Active	S3.LARGE8	172.16.0.100	55000	2	2020-07-03 15:47:21	0d 0h 0m 14s

# Cross-Region Disaster Recovery

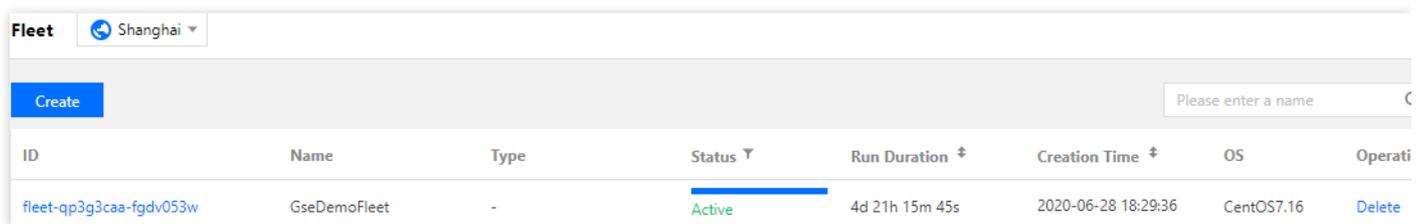
Last updated : 2020-08-04 11:33:32

## Overview

This document describes how to implement cross-region disaster recovery through a game server queue.

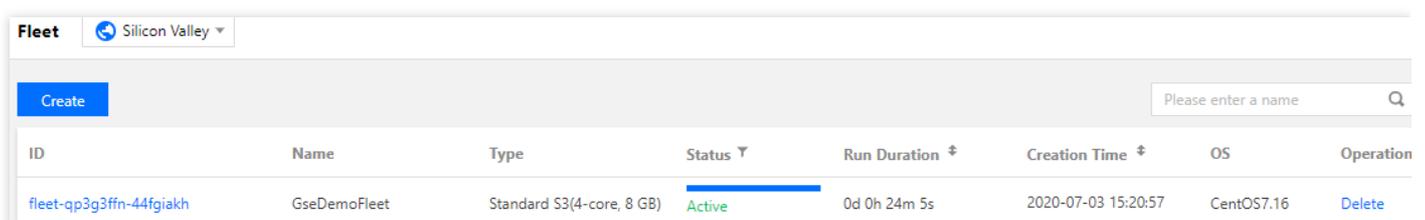
## Prerequisites

- Create two server fleets in Shanghai and Silicon Valley regions as instructed below:
  - Complete the first three steps in [Demo](#): click **Quick Upload of Demo Package**, **Quick Creation of Server Fleet**, and **Create Game Server Session** and then click **Complete**.
- You have created server fleet 1 (Shanghai region).



ID	Name	Type	Status	Run Duration	Creation Time	OS	Operation
fleet-qp3g3caa-fgdv053w	GseDemoFleet	-	Active	4d 21h 15m 45s	2020-06-28 18:29:36	CentOS7.16	Delete

- You have created server fleet 2 (US region).



ID	Name	Type	Status	Run Duration	Creation Time	OS	Operation
fleet-qp3g3ffn-44fgiakh	GseDemoFleet	Standard S3(4-core, 8 GB)	Active	0d 0h 24m 5s	2020-07-03 15:20:57	CentOS7.16	Delete

## Directions

### Creating game server queue

1. Log in to the [GSE Console](#) and click **Queue** on the left sidebar to enter the game server queue page.
2. Select the service region in the top-left corner and click **Create**.

3. In the game server queue creation page, enter the basic information:
  - Identifier: enter a valid identifier, which can contain letters only and is "disasterrecovery" in this example.
  - Timeout Allocation: enter the max time that a game server session request can be retained in a multi-region deployment. It can be up to 600 seconds and is 30 seconds in this example.
4. Enter the latency policy:
  - Use only one policy where the max player latency is specified as 150 ms to search for server fleets whose latency is up to 150 ms for any player.
5. Select the created server fleet 1 (Shanghai region) and server fleet 2 (US region) as the target.
6. Click **OK** to complete creating the game server queue.

**Basic Info**

Identifier ⓘ \*  ✓

Timeout Allocation ⓘ  sec ✓

**Latency Policy**

Priority	Time Consumed (s) ⓘ	Max Player Delay (ms) ⓘ	Operation
1	Remaining Timeout	<input type="text" value="150"/> ms	<a href="#">Delete</a>

[+ Add Latency Policy](#)

**Target**

Priority	Region	Type	ID and Name	Ope...
1 ↑ ↓	<input type="text" value="Shanghai"/>	<input type="text" value="Fleet"/>	<input type="text" value="fleet-..."/>	<a href="#">Delete</a>
2 ↑ ↓	<input type="text" value="Silicon Valley"/>	<input type="text" value="Fleet"/>	<input type="text" value="fleet-..."/>	<a href="#">Delete</a>

[+ Add Target](#)

## Requesting a server address as no failure occurs

Call the `StartGameServerSessionPlacement` TencentCloud API in the code to place the game server session in the server fleet process. This example uses [TencentCloud API Explorer](#) for quick creation.

### ⓘ Note :

Input parameter description:

- `Region` indicates the region, which is "ap-shanghai" (East China (Shanghai)) in this example;
- `PlacementId` indicates the unique ID of the game server session placement, which is 1 in this example;

- `GameServerSessionQueueName` indicates the game server session queue name, which is "disasterrecovery" in this example;
- `MaximumPlayerSessionCount` indicates the maximum number of concurrent players allowed by the game server to connect to the game session, which is 2 in this example;
- `DesiredPlayerSessions.N` indicates the player game session information, where `PlayerId` is the unique player ID associated with the player session. In this example, two values of 1 and 2 are entered respectively;
- `PlayerLatencies.N` indicates the player latency, where `PlayerId` is the player ID, `RegionIdentifier` is the name of the region where the latency occurs, and `LatencyInMilliseconds` is the latency in milliseconds. In this example, four value sets are entered, i.e., [1, ap-shanghai, 100], [1, na-siliconvalley, 50], [2, ap-shanghai, 60], and [2, na-siliconvalley, 80].

Service Management APIs

**StartGameServerSess...** 2019-11-12

**Input Parameters**

Region  View Only Required Parameters

ap-shanghai

PlacementId

GameServerSessionQueueName

MaximumPlayerSessionCount

DesiredPlayerSessions.N  (Optional)

1

PlayerId

PlayerData

Add

GameProperties.N  (Optional)

1

Key

Value

Add

Service Management APIs

**StartGameServerSess...** 2019-11-12

PlayerLatencies.N  (Optional)

1

PlayerId

RegionIdentifier

DescribeGameServerSessionPl	<input type="text" value="ap-shanghai"/>
DescribeGameServerSessions	LatencyInMilliseconds
DescribePlayerSessions	<input type="text" value="100"/>
GetGameServerSessionLogUrl	
GetInstanceAccess	2
JoinGameServerSession	PlayerId
SearchGameServerSessions	<input type="text" value="1"/>
<b>StartGameServerSessionPlac</b>	RegionIdentifier
StopGameServerSessionPlace	<input type="text" value="na-siliconvalley"/>
UpdateGameServerSession	LatencyInMilliseconds
	<input type="text" value="50"/>

and APIs

**Service Management APIs**

- CreateGameServerSession
- DescribeGameServerSessionD
- DescribeGameServerSessionPl
- DescribeGameServerSessions
- DescribePlayerSessions
- GetGameServerSessionLogUrl
- GetInstanceAccess
- JoinGameServerSession
- SearchGameServerSessions
- StartGameServerSessionPlac**
- StopGameServerSessionPlace
- UpdateGameServerSession

**StartGameServerSess...** 2019-11-

3	PlayerId	<input type="text" value="2"/>
	RegionIdentifier	<input type="text" value="ap-shanghai"/>
	LatencyInMilliseconds	<input type="text" value="60"/>
4	PlayerId	<input type="text" value="2"/>
	RegionIdentifier	<input type="text" value="na-siliconvalley"/>
	LatencyInMilliseconds	<input type="text" value="80"/>

[Add](#)

**Scheduling result evaluation of latency policy:**

Latency in two players' arrival at the target address:

- The latency for player 1 is 100 ms to Shanghai and 50 ms to Silicon Valley.
- The latency for player 2 is 60 ms to Shanghai and 80 ms to Silicon Valley.

As the latency policy specifies that only servers whose latency for any player is up to 150 ms can be matched and both the Silicon Valley and Shanghai regions meet the requirement, a game server session will be automatically created in server fleet 1 (Shanghai region) with a higher priority.

Server Fleet Details ( fleet-qp3g3caa-fgdv053w )								
<a href="#">Alarming Configuration</a> <a href="#">View Monitoring</a>								
Basic Info	Event	Instance List	Scaling	Game Server Session	Process Management	Ports and Protocol	Asset Package Info	VPC
Game Server Sessi...	Name	Status ▼	Instance Type	IP	Port	Player Session	Creation Time ↕	Run Duration ↕
qcs:gse:ap-s...		Active	S5.SMALL2	81.68.144.188	59213	1	2020-06-28 18:3...	4d 21h 37m 18s

## Automatic disaster recovery in case of failure

Suppose the Shanghai region fails and its speed cannot be tested.

### **Note :**

Input parameter description:

- `PlayerLatencies.N` indicates the player latency, where `PlayerId` is the player ID, `RegionIdentifier` is the name of the region where the latency occurs, and `LatencyInMilliseconds` is the latency in milliseconds. In this example, four value sets are entered, i.e., [1, ap-shanghai, 0], [1, na-siliconvalley, 50], [2, ap-shanghai, 0], and [2, na-siliconvalley, 80]. In case that the region speed cannot be tested, enter 0 or an infinite number for the latency value, or leave it empty. In this example, the latency to Shanghai is entered as 0.
- Keep the rest parameters the same as the ones above.



Service Management APIs

- CreateGameServerSession
- DescribeGameServerSessionD
- DescribeGameServerSessionPl
- DescribeGameServerSessions
- DescribePlayerSessions
- GetGameServerSessionLogUrl
- GetInstanceAccess
- JoinGameServerSession
- SearchGameServerSessions
- StartGameServerSessionPlac**
- StopGameServerSessionPlac
- UpdateGameServerSession

### StartGameServerSess...

2019-11-12

#### Input Parameters

Region  View Only Required Parameters

ap-shanghai

PlacementId [?](#)

1

GameServerSessionQueueName [?](#)

disasterrecovery

MaximumPlayerSessionCount [?](#)

2

DesiredPlayerSessions.N [?](#) (Optional)

1

PlayerId

1

PlayerData

string

Add

GameProperties.N [?](#) (Optional)

1

Key

2

Value

string

Add

Service Management APIs

- CreateGameServerSession
- DescribeGameServerSessionD

### StartGameServerSess...

2019-11-12

string

PlayerLatencies.N [?](#) (Optional)

1

PlayerId

1

DescribeGameServerSessionPl	RegionIdentifier	ap-shanghai
DescribeGameServerSessions	LatencyInMilliseconds	0
DescribePlayerSessions		
GetGameServerSessionLogUrl		
GetInstanceAccess	2	
JoinGameServerSession	PlayerId	1
SearchGameServerSessions	RegionIdentifier	na-siliconvalley
<b>StartGameServerSessionPlac</b>	LatencyInMilliseconds	50
StopGameServerSessionPlac		
UpdateGameServerSession		

and APIs

**StartGameServerSess...** 2019-1

**Service Management APIs**

- CreateGameServerSession
- DescribeGameServerSessionD
- DescribeGameServerSessionPl
- DescribeGameServerSessions
- DescribePlayerSessions
- GetGameServerSessionLogUrl
- GetInstanceAccess
- JoinGameServerSession
- SearchGameServerSessions
- StartGameServerSessionPlac**
- StopGameServerSessionPlac
- UpdateGameServerSession

3	PlayerId	2
	RegionIdentifier	ap-shanghai
	LatencyInMilliseconds	0
4	PlayerId	2
	RegionIdentifier	na-siliconvalley
	LatencyInMilliseconds	80

[Add](#)

### Scheduling result evaluation of latency policy:

Latency in two players' arrival at the target address:

- The latency for player 1 is 0 ms to Shanghai and 50 ms to Silicon Valley.
- The latency for player 2 is 0 ms to Shanghai and 80 ms to Silicon Valley.

A latency of 0 ms to the Shanghai region indicates that the latency cannot be measured due to a failure in Shanghai; therefore, a game server session will be automatically created in server fleet 2 in the US region.

← **Server Fleet Details ( fleet-qp3g3caa-fgdv053w(G) )** [Alarming Configuration](#) [View Monitor](#)

Basic Info   Event   Instance List   Scaling   **Game Server Session**   Process Management   Ports and Protocol   Asset Package Info   VPC

Game Server Sessi...	Name	Status	Instance Type	IP	Port	Player Session	Creation Ti...	Run Durati...
qcs:gse:na-si...	-	Active	S3.LARGE8	172.16.0.100	5000	0	2020-07-03 15:...	0d 0h 22m 2s

### Manual disaster recovery in case of failure

If a region fails, you need to manually remove server fleets in it from the target list in the game server queue, and GSE will schedule game server sessions to the remaining server fleets in the target list.

**Basic Info**

Identifier ⓘ testw

Timeout Allocation ⓘ  sec

**Latency Policy**

Priority	Time Consumed (s) ⓘ	Max Player Delay (ms) ⓘ	Operation
1	Remaining Timeout	<input type="text" value="150"/> ms	Delete

[+ Add Latency Policy](#)

**Target**

Priority	Region	Type	ID and Name	Opera...
1 ↑ ↓	Shanghai	Fleet	fleet-qp3g3caa-fgdv053w(G)	Delete
2 ↑ ↓	Silicon Valley	Fleet	fleet-qp3g3ffn-44fgiakh(Gse)	Delete

[+ Add Target](#)