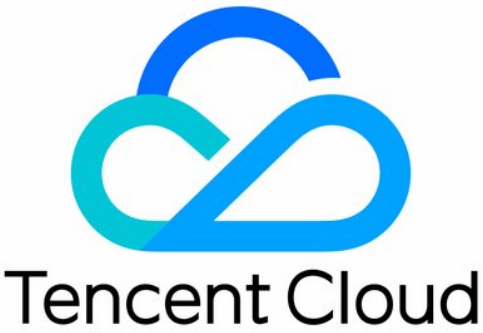
Game Server Elastic-scaling

Operation Guide

Product Documentation



Contents

Operation Guide
Creating Asset Package

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

Overview

This document describes how to create an asset package, which will be uploaded to and deployed in GSE for game hosting.

Directions

1. Log in to the GSE console and click **Asset** on the left sidebar.
2. Click **Create** in the top-left corner.
3. To create an asset package, you can upload one through the console page or CLI.

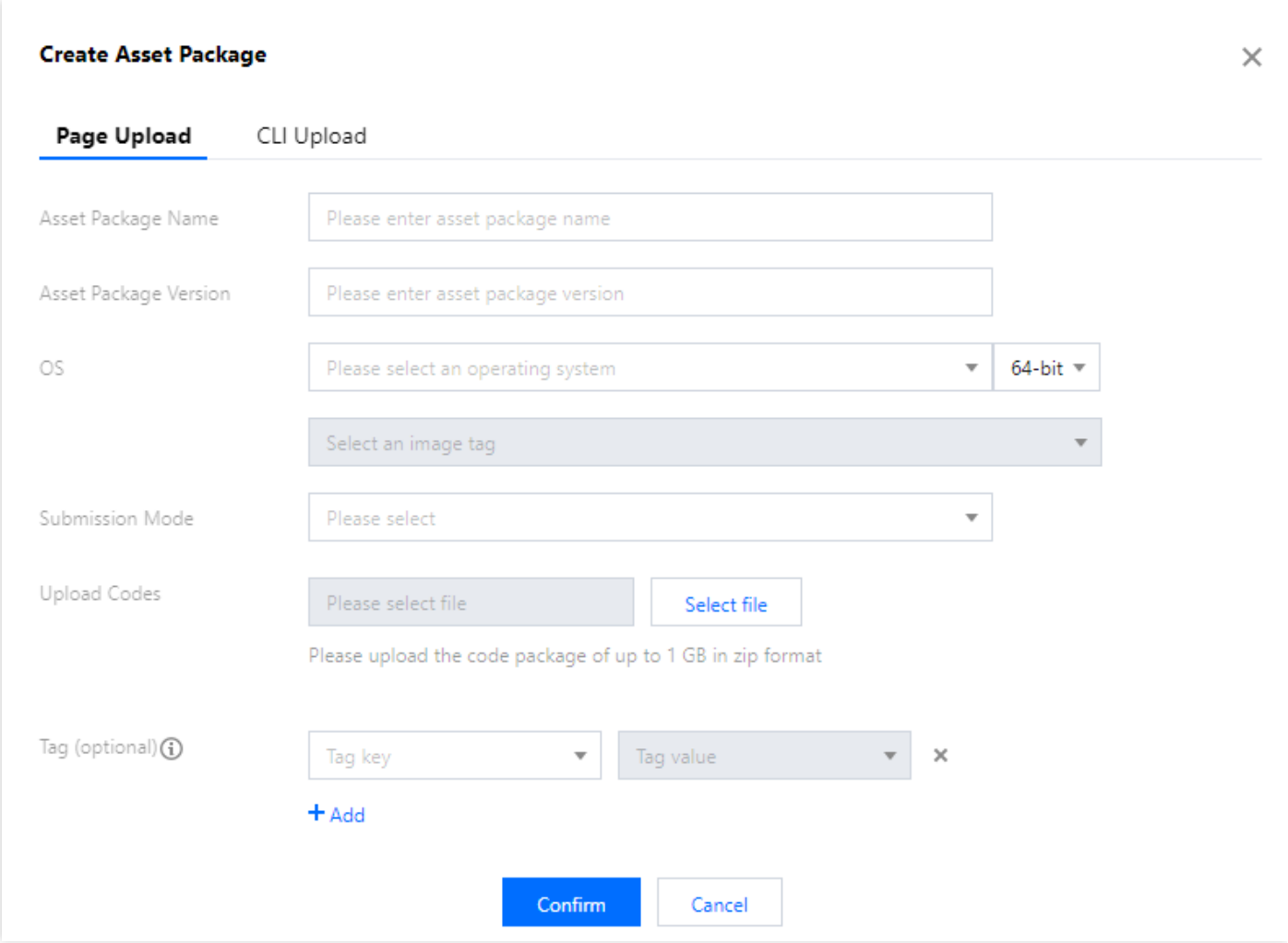
Page upload

Enter the basic package information, including the package name and version. Select the OS and submission mode, upload the code, and click **OK**.

- Asset Package Name: descriptive name of the asset package to be created, which does not need to be unique and can be modified and updated.
- Asset Package Version: detailed information of asset package version, which is used to distinguish between different asset package versions.
- OS: the operating system on which the game server asset package runs. It cannot be modified after the asset package is created.
- Submission Mode: currently, only local zip packages can be uploaded.
- Upload Codes: the asset package requires integrating the gRPC framework. The asset package directory must contain all the components necessary for running your game server, including executable files, dependency packages, and installation scripts of the game server. All are compressed into a zip package.
  - Game server executable file: a file required for running the game server. The asset package can contain multiple executable files, but only if they are built for the same platform.
  - Dependency package: any file required for running the game server executable file.
  - Installation script: executes the tasks for installing the asset package on the GSE-hosted server. This file must be placed in the root directory of the asset package and will run once during fleet

creation. For more information, see Game Process Launch Configuration.

- 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.



CLI upload

Step 1: download the script

Click here to download the `uploadasset.py` script.

Step 2: install the script dependency packages

Running the script requires the following two dependency packages.

- **tencentcloud-sdk-python**
  - Installation command: `pip install tencentcloud-sdk-python` .
  - Supported version: TencentCloud API SDK Python V2.7-V3.6. For more information, see Python SDK.
- **cos-python-sdk-v5**

- Installation command: `pip install -U cos-python-sdk-v5` .
- Supported version: For more information, see Python SDK.

Step 3: run the script

1. Modify the configuration

Configure your authentication in the script. Replace `secret_id` and `secret_id` with your actual values.

2. Execute the command

Execute the following command to run the script:

```
python uploadasset.py --local_path ./game_folder/
```

You will see `help` after running the command code. Click `help` to view more parameter information.

Note :

1. When you use the CLI to upload the asset package, you can configure the parameter `--local_path` to specify the upload folder path. The script will automatically compress and package the content under this path (only the contents under the path rather than the folder are packaged).
2. The operating system and bit can be uploaded. The following methods are recommended:

```
python uploadasset.py --local_path ./game_folder/ --os_type=Windows --os_bit=64
python uploadasset.py --local_path ./game_folder/ --os_type=CentOS --os_bit=64
python uploadasset.py --local_path ./game_folder/ --image_id=img-9qabwvbn
```

Creating Image Resource

Last updated : 2021-05-18 14:20:49

This document describes how to use image resources to create a server fleet.

Prerequisites

- You already have a Cloud Virtual Machine (CVM) instance.

Directions

1 Upload game executable programs to the CVM instance under the path `/local/game/` .
2. Log in to the CVM console and click **Instances** on the left sidebar to enter the instance management page.
3. On the instance management page, select the instance and create image. For more information, see Creating Custom Images.

Server Fleet
Creating Server Fleets

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

Overview

This document describes how to create a server fleet, which is a group of managed resources in the form of CVM instances, to deploy game servers. The fleet size is subject to the number of instances you assign, and can be manually or automatically scaled to meet players' needs.

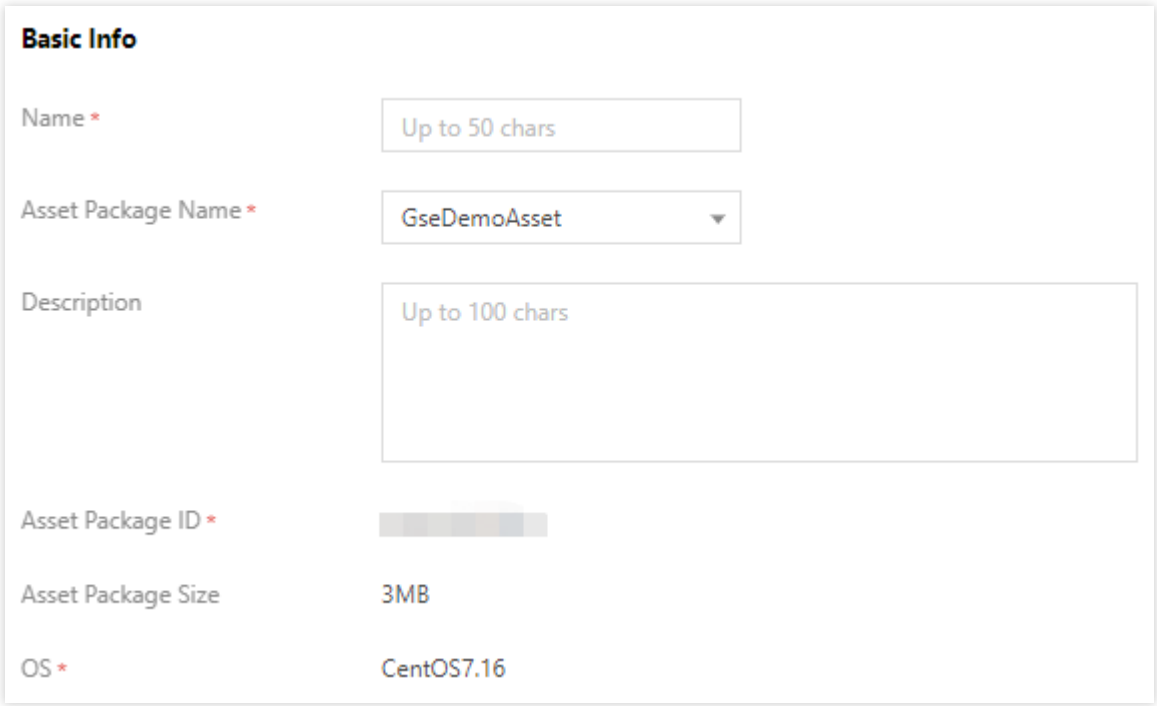
Directions

1. Log in to the GSE console and click **Fleet** in the left sidebar to enter the server fleet page.
2. Select the service region at the top-left corner and click **Create**.
3. On the **Create Fleet** page, enter information such as basic info, process management, deployment configuration, and internet.

Basic information

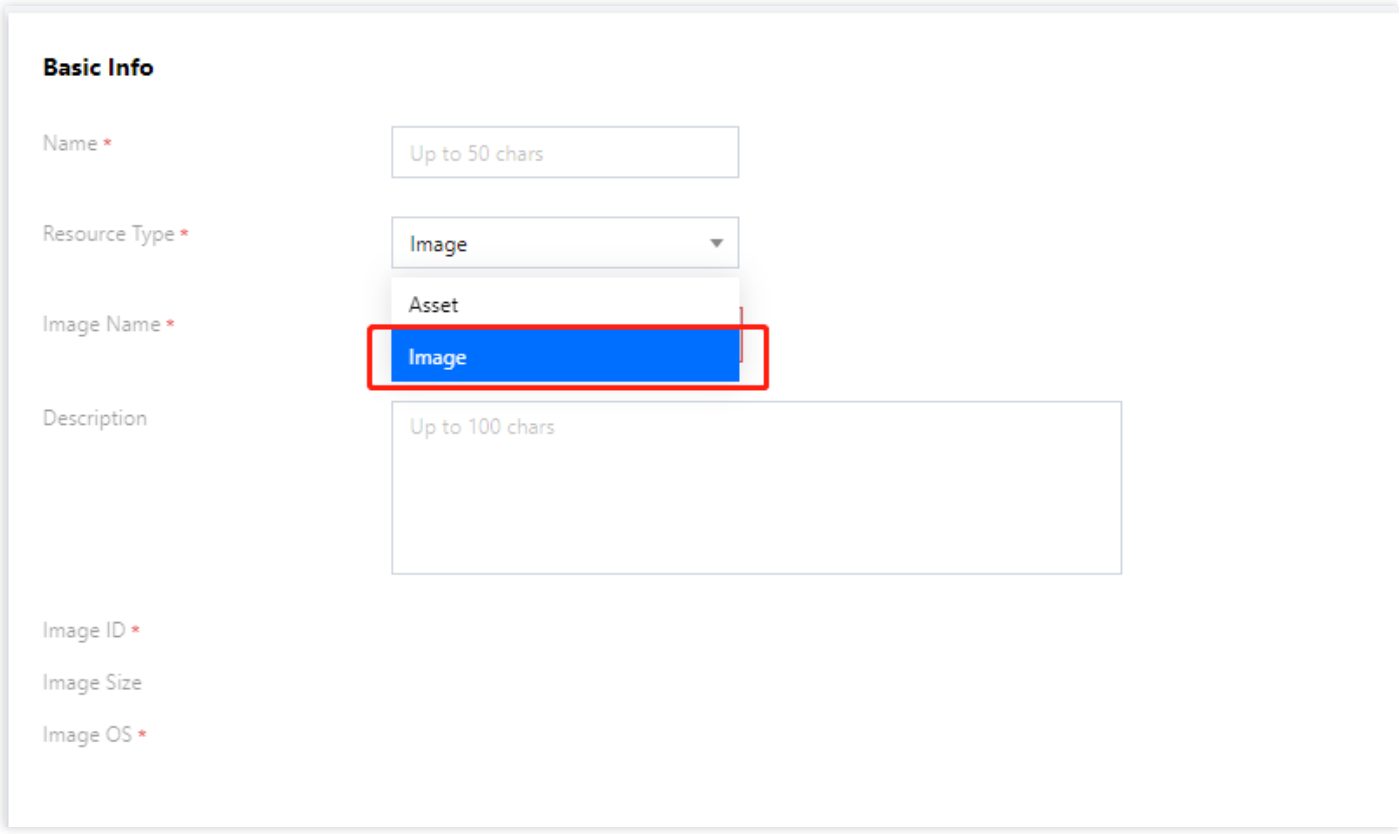
To create a server fleet, you can use **asset package** or **image** according to the resource type:

- **Asset package**



  - Name: enter the server fleet name. We recommend you use a meaningful fleet name, so that it can be easily identified in the list.
  - Resource Type: select **Asset Package** from the drop-down list.
  - Asset Package Name: select a valid asset package from the drop-down list or create one first.
  - Description: enter the server fleet description, which is optional and used to help identify the fleet.
  - **Asset Package ID**, **Asset Package Size**, and **OS** will be automatically entered based on the selected asset package name.

- **Image**



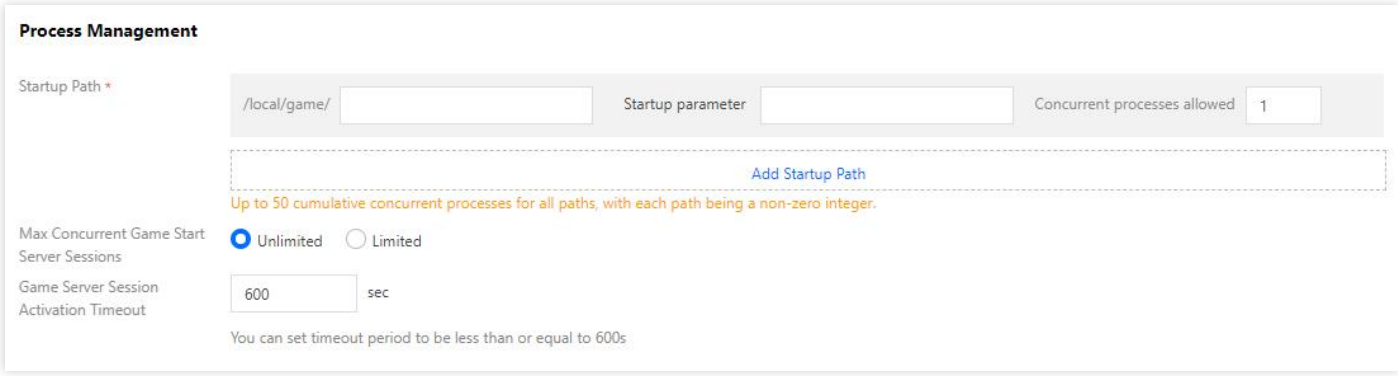
  - Name: enter the server fleet name. We recommend you use a meaningful fleet name, so that it can be easily identified in the list.
    Resource Type: select **Image** from the drop-down list.
  - Image Name: select a valid image from the drop-down list or create one first. For more information, see Creating an Image.
  - Description: enter the server fleet description, which is optional and used to help identify the fleet.
  - **Image ID**, **Image Size** and **Image OS** will be automatically entered based on the selected image name.

> Note :
> Using **Image** to create a server fleet requires authorization, which means you need to authorize the image resources to GSE. To do so, click **Share** > **Go to CAM** > **Authorize**.

Process management

Configure how server processes run on each CVM instance.

- Launch Path: enter the path of a server executable file in the asset package. Download the asset package to the `/local/game/` path on the Linux platform and to the `C:¥game¥` path on the Windows platform. The decompressed full path of the server process is as follows
- Liunx: `/local/game/YourGameServerBuild`
- Windows: `C:¥game¥YourGameServerBuild.exe`
- Launch Parameter: enter the information to be delivered to the server executable file during launch. It is in the format of a set of command line parameters and is optional.
- Concurrent processes allowed: specify this configuration allows how many processes concurrently run on each CVM instance in the server fleet.
- Max Concurrent Game Server Session Activation: set the number of game server sessions that can be concurrently activated on a CVM instance. You can select **Unlimited** or **Limited** (the maximum value is 20,000). When multiple new game server sessions are launched on one CVM instance, this limit can reduce their performance impact on each other.
- Game Server Session Activation Timeout: enter the maximum time period for a new game server session to be activated. You can set the timeout period to be no more than 600s.

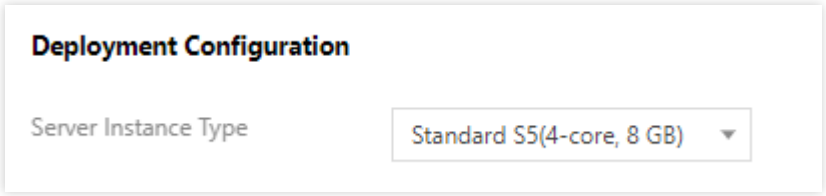


Note :

- Concurrent processes allowed: the total number of game processes that need to be launched for a launch path binary
- Max Concurrent Game Server Session Activation: the number of instantaneous concurrencies of the game server sessions in “activating” state.

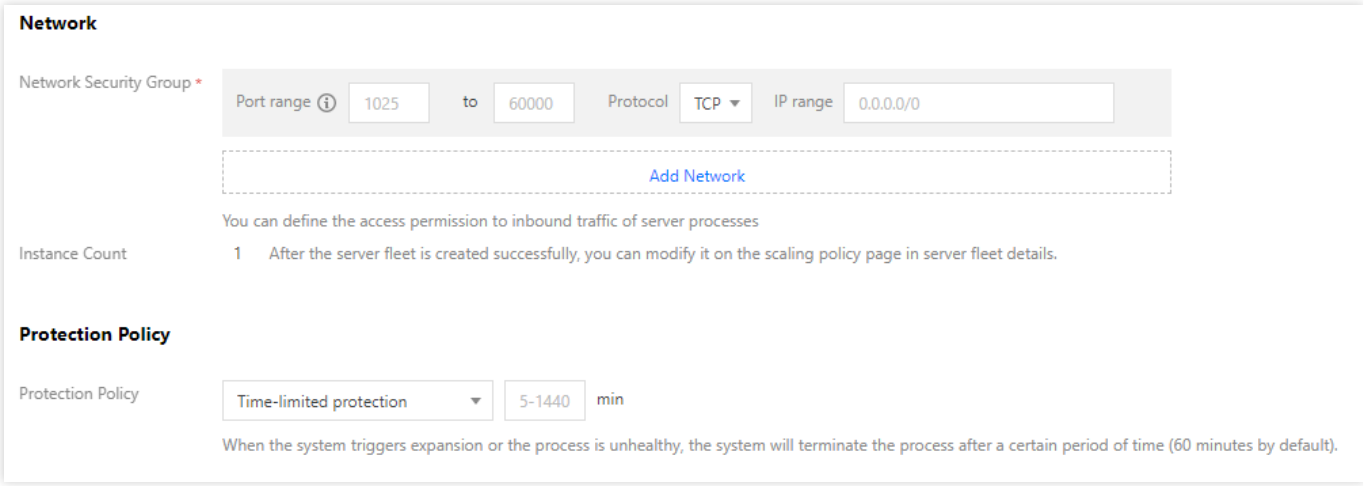
Deployment configuration

- Server Instance Type: server model of the server fleet to be created.
- Enable VPC: after enabling access to Tencent Cloud VPC, you can access the servers and other resources in your VPC.



Internet

- Network Security Group: you can define the access permission to inbound traffic of your server processes. Before allowing access, you must set at least one port for the fleet.
  - Port range: specify the range of numbers of the ports that can be opened for inbound connections to the game servers. The range must be 1025–60000.
  - Protocol: select TCP or UDP as the communication protocol of the fleet.
  - IP range: specify the valid IP range for CVM instances in the fleet. You can use a CIDR block to represent a range (such as `0.0.0.0/0`, which indicates that access requests from any user can be allowed).
- Number of instances: refers to the number of CVM instances. After the server fleet is created successfully, this number can be modified on the scaling policy page in the server fleet details.
- Protection Policy: includes time-period protection, full protection, and no protection.
  When you call the process termination API, the process will be actively terminated. If you do not call this API, one of the following protection policies will be executed accordingly:
  - Time-period protection: in case of a scaling-in or an unhealthy process, the system will terminate the process after a period of time between 5 to 1,440 minutes (60 minutes by default).
  - Full protection: the process can be terminated only when there are no processes running on the CVM instance.
  - No protection: in case of a scaling-in or an unhealthy process, the system will terminate the process after up to 5 minutes.



4. After configuring the information above, click **Create** to create a server fleet.
5. After the server fleet is successfully created, you can view, delete, or perform other operations on the server fleet page. You can also click the server fleet ID to view information such as basic information, events, instance list, scaling, game sessions, process management, ports and protocols, asset packages, and VPC.

Viewing Event

Last updated : 2021-03-30 10:41:23

Overview

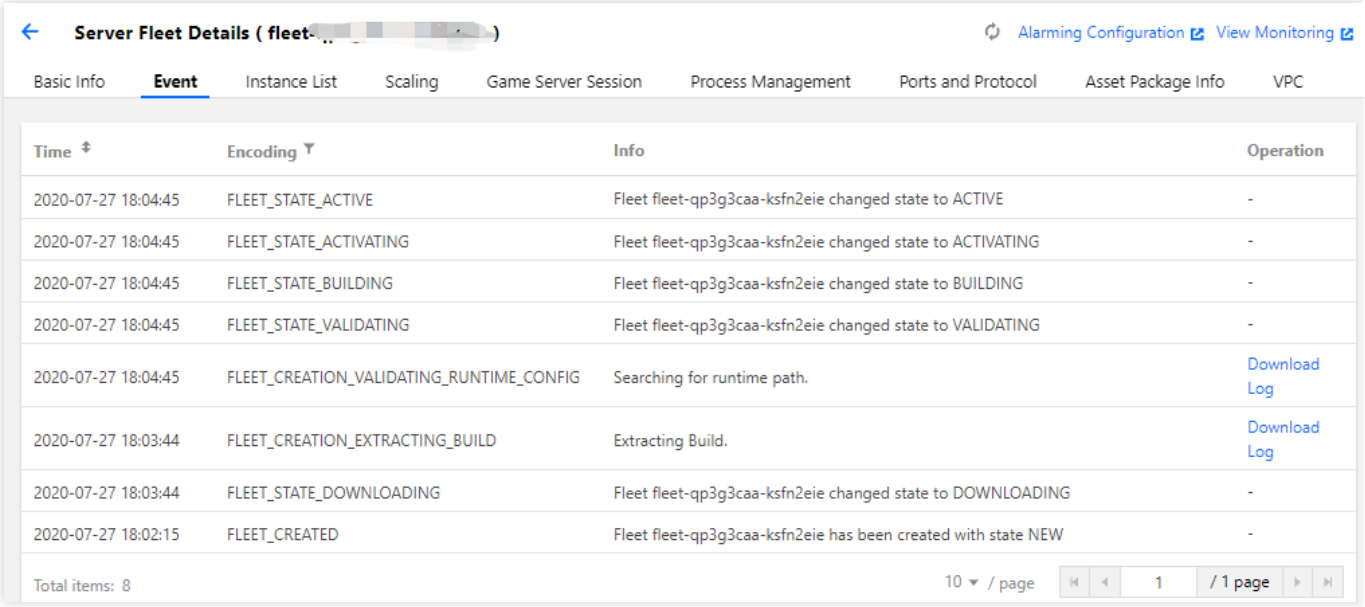
This document describes how to view events for activity tracing, troubleshooting, and debugging. When each stage of server fleet creation is completed, a series of events related to the fleet and current fleet status will be generated. You can trace all these events in the console.

Prerequisites

You have created server fleets.

Directions

1. Log in to the GSE Console and create a server fleet. For more information, please see Creating Server Fleets.
2. Click the **ID** of the created server fleet to enter the fleet details page. Click the **Event** tab to enter the event page.
   The event details list contains the following information: time, code, message, and operation.
   - **Time**: event occurrence time in the format of `yyyy-mm-dd hh:mm:ss` .
   - **Code**: event type. For specific event types, please see the relevant description below.
   - **Message**: event message.
   - **Operation**: download log.
   - **Event type**: currently, there are deployment and creation events, VPC peering connection events, and other fleet events.



Deployment and creation events

Code	Description
FLEET_CREATED	A fleet has been successfully created with status NEW. The event message contains the fleet ID
FLEET_STATE_DOWNLOADING-FLEET	The status has changed from NEW to DOWNLOADING, and the asset package is being downloaded onto the fleet instances for installation
FLEET_BINARY_DOWNLOAD_FAILED	Unable to download the asset package onto the fleet instances
FLEET_CREATION_EXTRACTING_ASSET	The asset package has been successfully downloaded onto the instances. The asset package files have been extracted from the uploaded zip file and stored onto the instances. If this stage fails, the fleet cannot enter the ACTIVE status. The logs of this stage display the extracted instance list and are stored on the instances. You can access the logs using the URL in **PreSignedLogUrl**
FLEET_CREATION_RUNNING_INSTALLER	The asset package has been successfully extracted, and GSE is running the built

	installation script (if any). If this stage fails, the fleet cannot enter the ACTIVE status. The logs of this stage list the installation steps and whether the installation has been successfully completed. You can access the logs using the URL in **PreSignedLogUrl**
FLEET_CREATION_VALIDATING_RUNTIME_CONFIG	The building process has been successfully completed, and GSE is verifying the game server launch paths specified in the runtime configuration of the fleet. If a launch path is listed, GSE will try launching the game server process and wait for the process to report its readiness. If this stage fails, the fleet cannot enter the ACTIVE status. The logs of this stage list the launch paths in the runtime configuration and whether each launch path has been found. You can access the logs using the URL in **PreSignedLogUrl**
FLEET_STATE_VALIDATING- FLEET	The status has changed from DOWNLOADING to VALIDATING
FLEET_VALIDATION_LAUNCH_PATH_NOT_FOUND	Runtime configuration verification failed, as the executable file specified in the launch path does not exist on the instance
FLEET_STATE_ASSETING- FLEET	The status has changed from VALIDATING to ASSETING
FLEET_VALIDATION_EXECUTABLE_RUNTIME_FAILURE	Runtime configuration verification failed, as the executable file specified in the launch path cannot run on the fleet instance
FLEET_STATE_ACTIVATING- FLEET	The fleet status has changed from ASSETING to ACTIVATING
FLEET_ACTIVATION_FAILED- FLEET	A step in fleet activation failed. This event code indicates that the asset package has been successfully downloaded onto the fleet instances,

	built, and verified, but the server process cannot be launched
FLEET_STATE_ACTIVE-FLEET	The status has changed from ACTIVATING to ACTIVE

VPC peering connection events

Code	Description
FLEET_VPC_PEERING_SUCCEEDED	A peering connection has been established between the VPCs of the GSE fleet and the VPC in your Tencent Cloud account
FLEET_VPC_PEERING_FAILED	The requested VPC peering connection failed. For more information, please see the details and status information of the event. A common cause is that two VPCs have overlapping CIDR blocks of IPv4 addresses. To fix this problem, please change the CIDR block of the VPC under your Tencent Cloud account
FLEET_VPC_PEERING_DELETED-VPC	The peering connection has been successfully deleted

Other fleet events

Code	Description
FLEET_SCALING_EVENT	The fleet capacity settings (number of required instances and expansion/reduction limits) have been changed. The event message contains the new capacity settings
FLEET_NEW_GAME_SESSION_PROTECTION_POLICY_UPDATED	The settings of the game server session protection policy of the fleet have been changed. The event message contains the old and new policy settings
FLEET_DELETED	A fleet deletion request was initiated
GENERIC_EVENT	An unspecified event occurred

FLEET_STATE_ERROR	Status failure
FLEET_INITIALIZATION_FAILED	Initialization failed
FLEET_VALIDATION_TIMED_OUT	Verification timed out
FLEET_ACTIVATION_FAILED_NO_INSTANCES	No instance
SERVER_PROCESS_INVALID_PATH	Invalid process path
SERVER_PROCESS_SDK_INITIALIZATION_TIMEOUT	Process SDK initialization timed out
SERVER_PROCESS_PROCESS_READY_TIMEOUT	Process preparation timed out
SERVER_PROCESS_CRASHED	The process crashed
SERVER_PROCESS_TERMINATED_UNHEALTHY	The process is unhealthy
SERVER_PROCESS_FORCE_TERMINATED	The process was forcibly terminated
SERVER_PROCESS_PROCESS_EXIT_TIMEOUT	Process exit timed out
GAME_SESSION_ACTIVATION_TIMEOUT	Game server session activation timed out
SERVER_PROCESS_PULL_FAILED	Failed to pull the process
SERVER_DOWN	The server is down

Viewing Instance List

Last updated : 2021-03-30 10:39:54

Overview

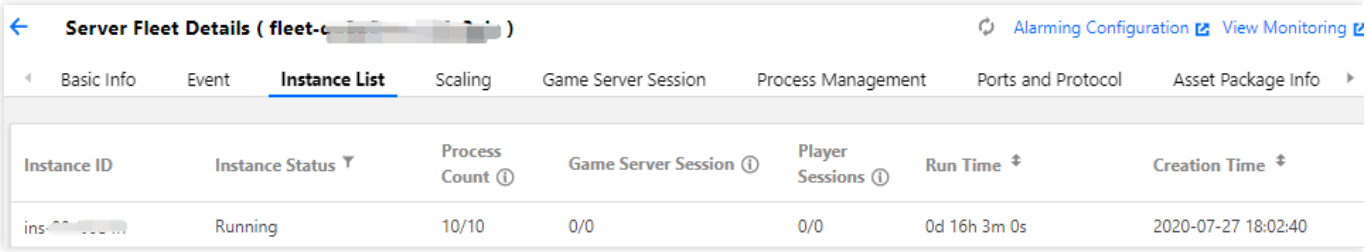
This document describes how to view all information about the instance list, including usage and metrics, through the console.

Prerequisites

You have created server fleets.

Directions

1. Log in to the GSE Console and create a server fleet. For more information, please see Creating Server Fleets.
2. Click the **ID** of the created server fleet to enter the fleet details page. Click the **Instance List** tab to enter the instance list page. The fields are as detailed below:
   - **Instance ID**: default ID generated by the system, which is up to 50 characters.
   - **Instance Status**: CVM instance status. Only "running" instances will be assigned to game server fleets.
   - **Process Count**: number of healthy/active processes.
   - **Game Server Session**: current/max game server sessions. If this metric is configured to be displayed, the first number indicates the number of currently active sessions, and the second number indicates the maximum number of sessions, such as 1/100; otherwise, "-" will be displayed.
   - **Player Session**: current/max player sessions. If this metric is configured to be displayed, the first number indicates the number of current sessions, and the second number indicates the maximum number of sessions, such as 1/100; otherwise, "-" will be displayed.
   - **Run Time**: duration between running start time and current time in the format of `xxd xxh xxm xxs` . The entries can be sorted in ascending/descending order.
   - **Creation Time**: creation time in the format of `yyyy-mm-dd hh:mm:ss` . The entries can be sorted in ascending/descending order.



Configuring Scaling

Last updated : 2021-03-30 10:35:14

Overview

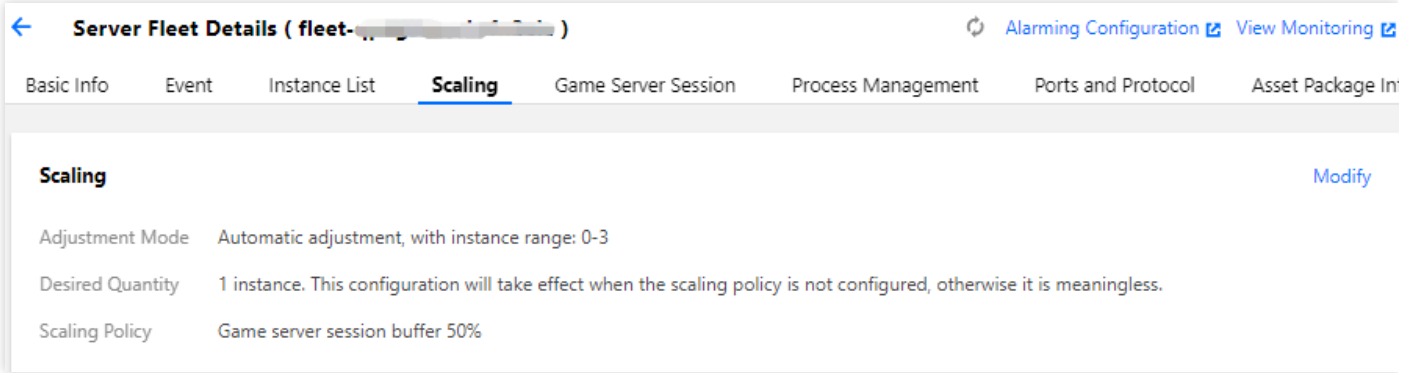
This document describes how to configure a scaling policy after a server fleet is successfully deployed. The size of a server fleet is subject to the number of instances contained in it. You can also set the lower and upper limits based on the actual needs of your game. No matter whether you choose manual or automatic adjustment, the target capacity in any server fleet capacity change requests cannot be out of the current limit range.

Prerequisites

You have created server fleets.

Directions

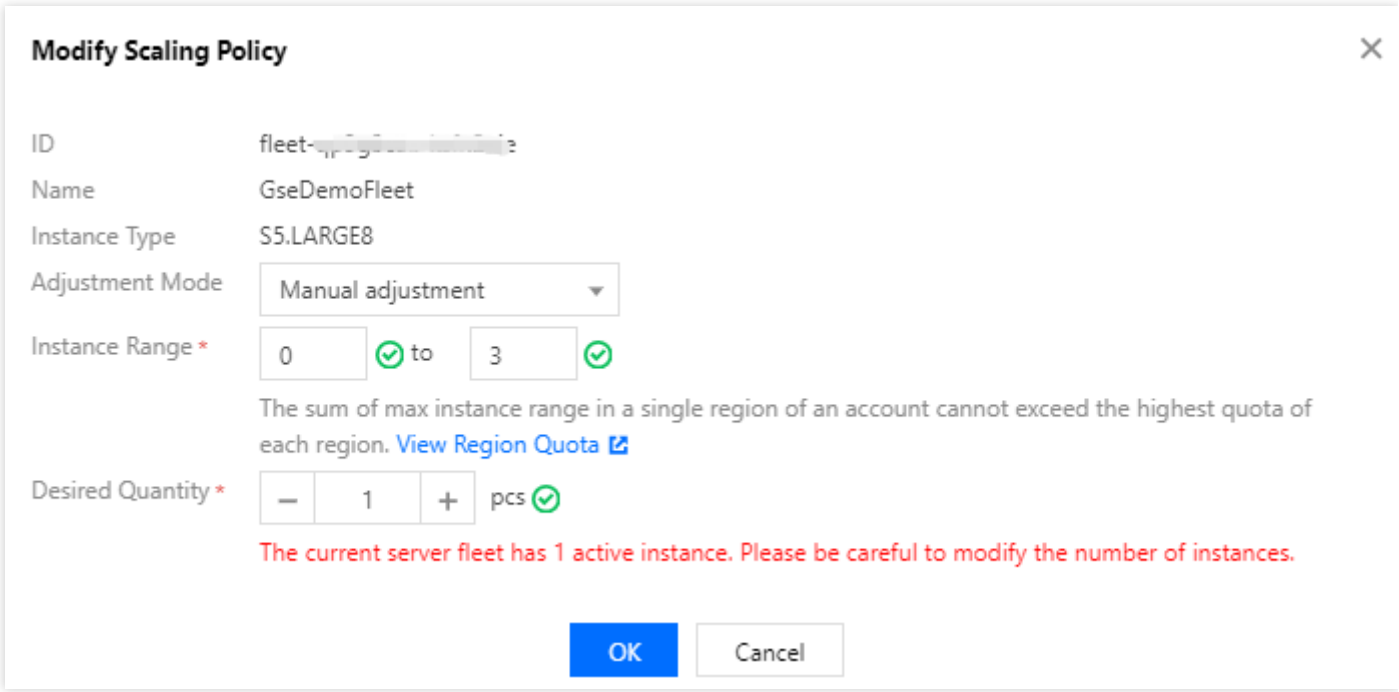
1. Log in to the GSE Console and create a server fleet. For more information, please see Creating Server Fleets.
2. Click the **ID** of the created server fleet to enter the fleet details page. Click the **Scaling** tab to enter the expansion/reduction page.
3. Click **Modify** to enter the scaling policy modification page.



The adjustment modes of a scaling policy include manual adjustment and automatic adjustment.

Manual adjustment

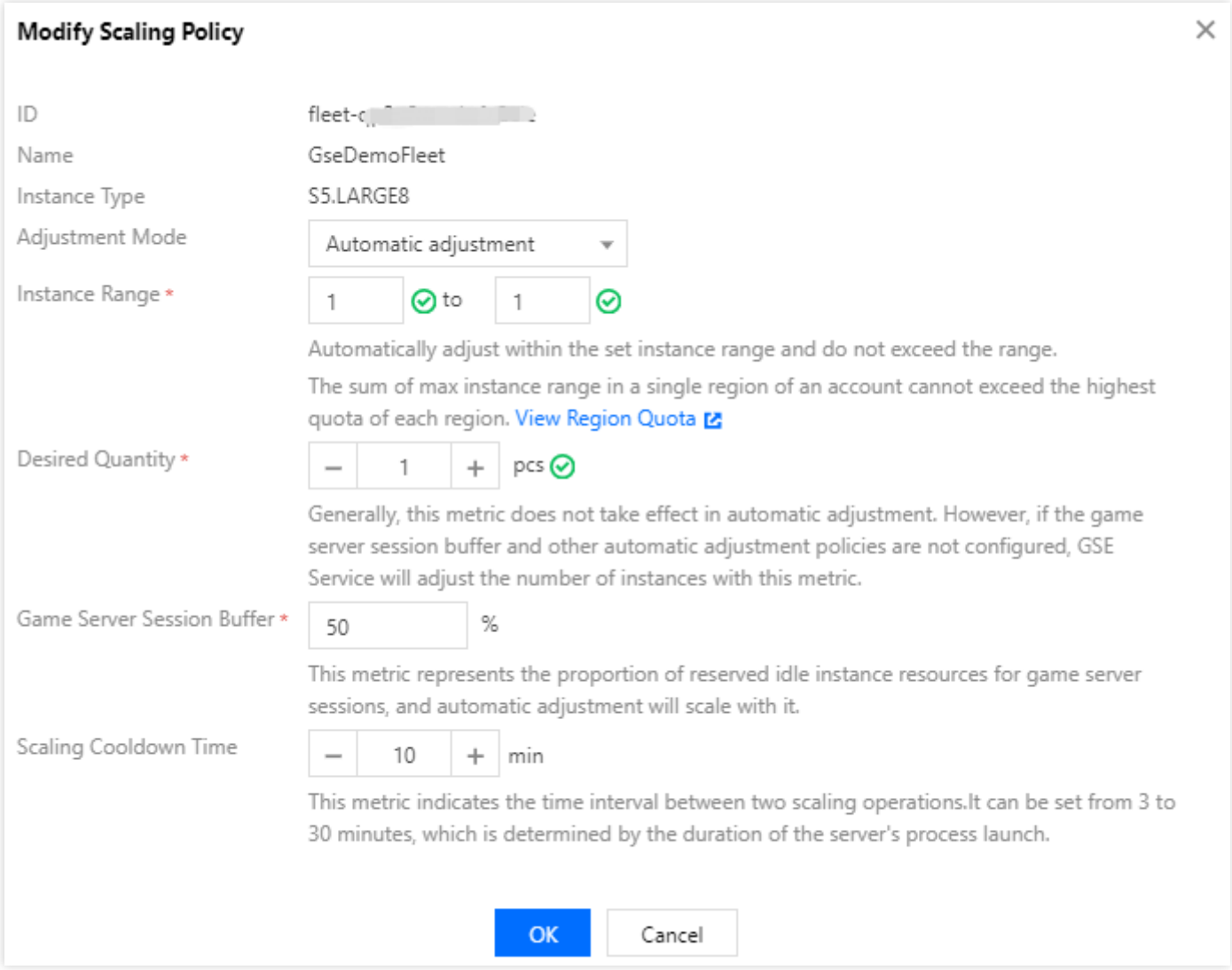
- Instance Range: it is in the format of minimum number of instances–maximum number of instances and is in integer type. The desired quantity should not exceed the specified instance range. If the first deployment of asset package fails, you can set this field to 0–1 only.
- Desired Quantity: it takes effect for manual adjustment. After being configured, it will become the only metric based on which the instance quantity will be adjusted.



Automatic adjustment

- Instance Range: it is in the format of minimum number of instances–maximum number of instances and is in integer type. Automatic adjustment cannot exceed the specified instance range. In each account, the max number of instances for each fleet combined in a single region cannot exceed the regional quota. You can view regional quotas here. To increase the CVM instance quota, please submit a ticket for application. You cannot set this field if the first deployment of asset package fails.
- Desired Quantity: 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, the system will adjust the number of instances with this metric.
- Game Server Session Buffer: this metric represents the proportion of reserved idle resources of the game server session, and automatic adjustment will be performed for scaling with it. It is an integer ranging from 1 to 99 in %. For example, if it is configured as 15%, expansion will start when the server resource utilization reaches 85%.
- Scaling Cooldown Time: 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.



Scheduled Update Plan for Scaling Policy

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

Overview

This document describes how to configure the scheduled update plan for auto-scaling policy after you deploy the server fleet successfully.

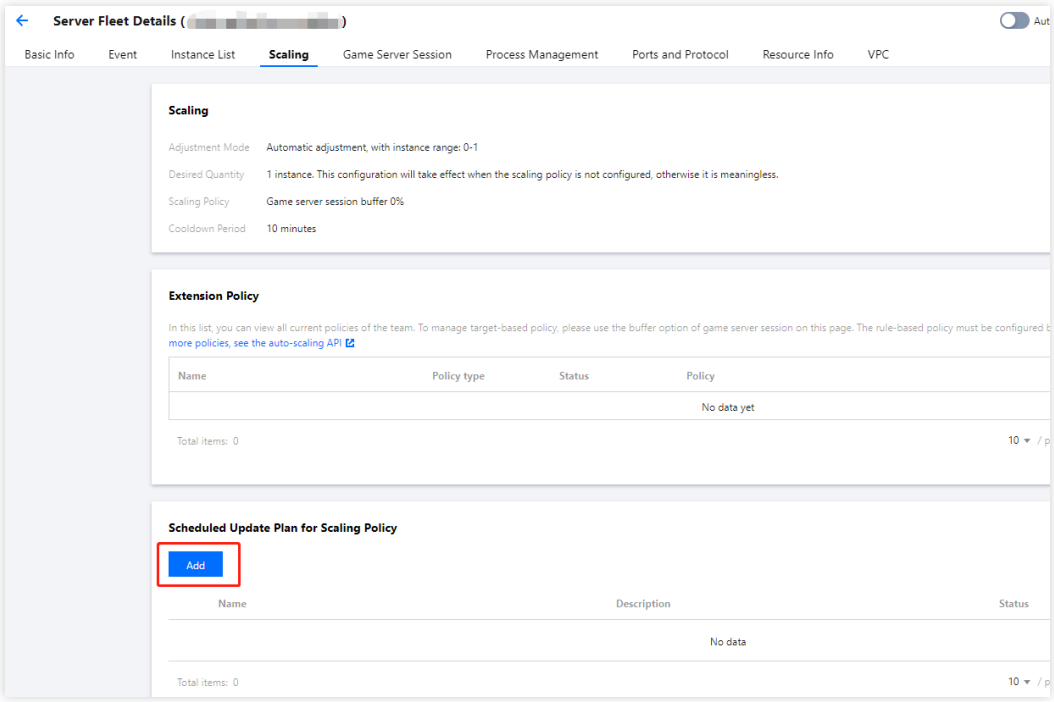
Prerequisites

You have created a server fleet.

Directions

1. Log in to the GSE console and click **Fleet** on the left sidebar.
2. Click the **ID** of the created server fleet to enter the fleet details page. Click the **Scaling** tab to enter the scaling page.

3. Click **Add** in the **Scheduled Update Plan for Scaling Policy** section.

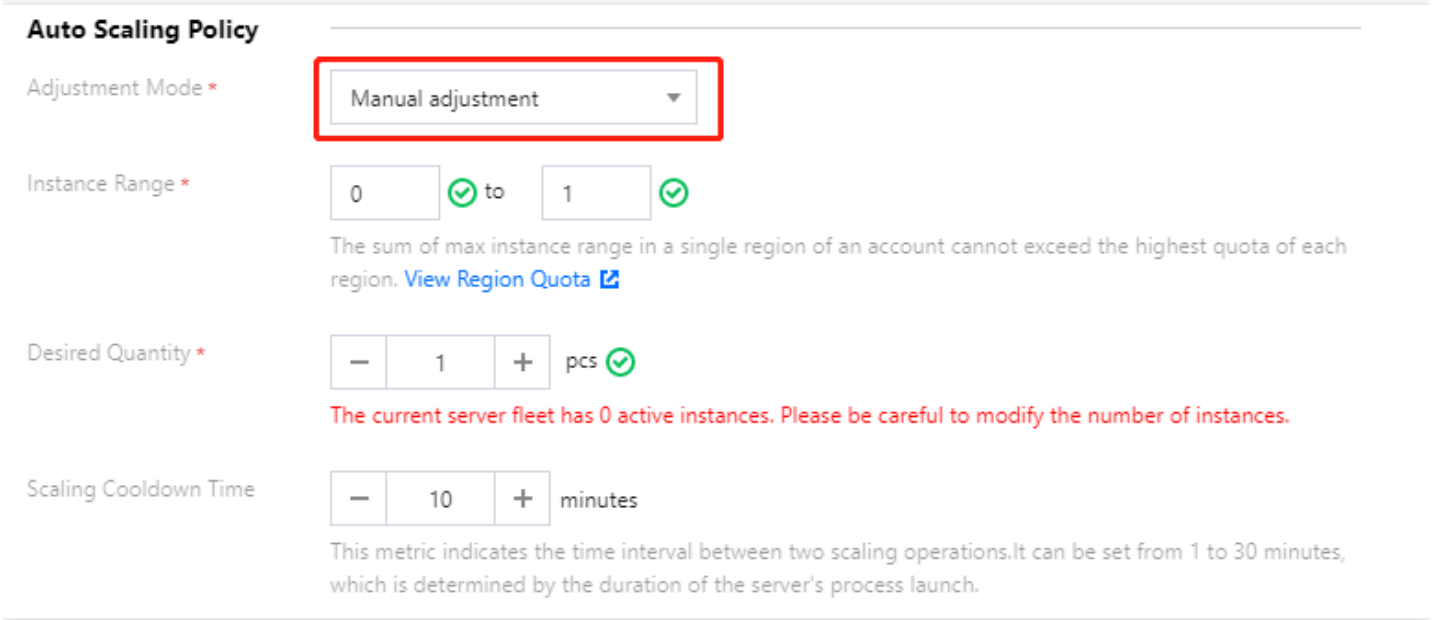


The scheduled update plan for scaling policy includes the setting of auto scaling policy and repeat cycle.

Auto scaling policy

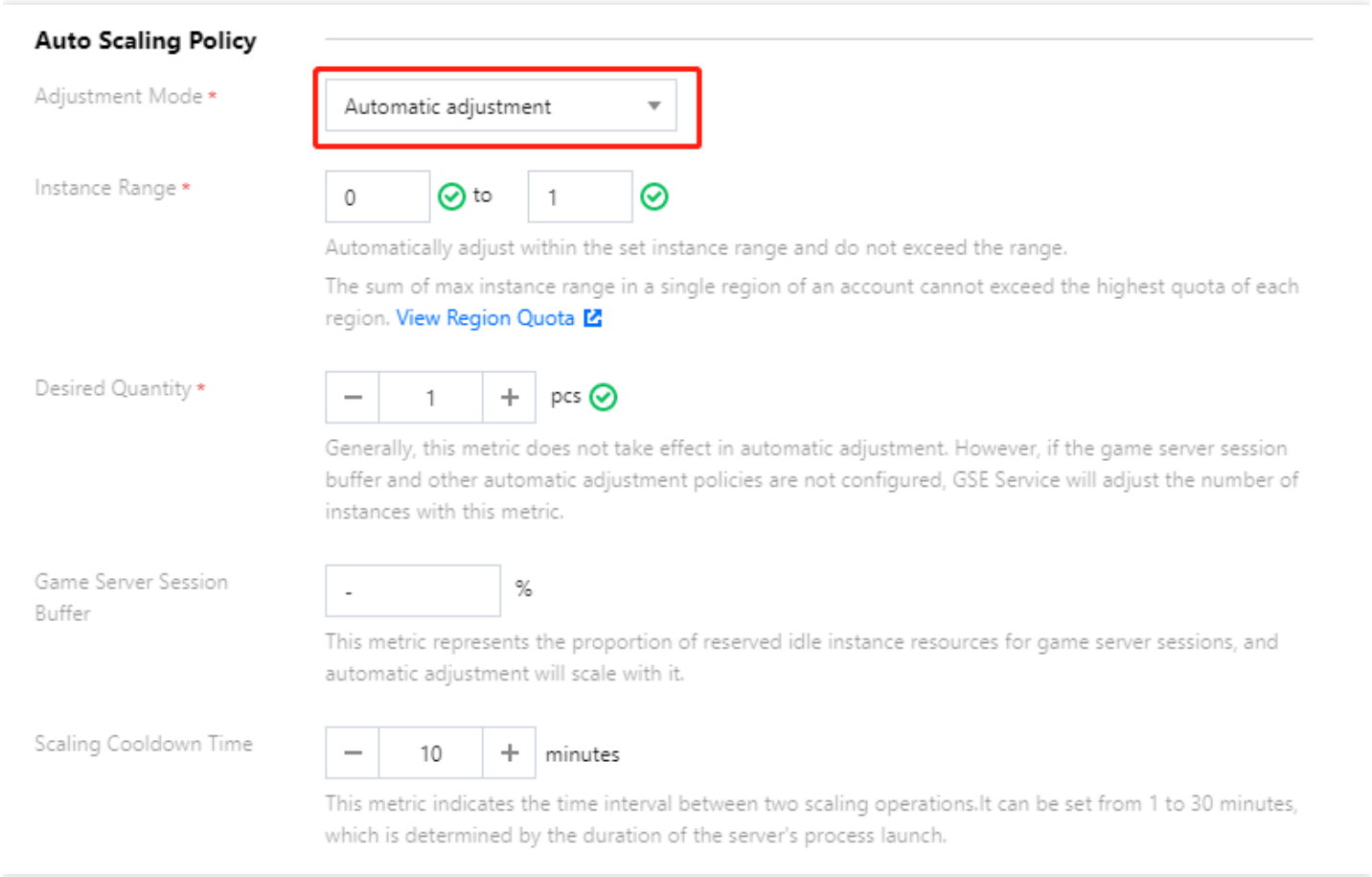
The adjustment mode of the auto scaling policy includes automatic adjustment and manual adjustment.

Manual adjustment



- **Instance Range**: it is in the format of minimum number of instances – maximum number of instances and is in integer type. The desired quantity should not exceed the specified instance range. If the first deployment of asset package fails, you can set this field to 0–1 only.
- **Desired Quantity**: it takes effect for manual adjustment. After being configured, it will become the only metric based on which the instance quantity will be adjusted.
- **Scaling Cooldown Time**: indicates the time interval between two scaling operations. It can be set from 1 to 30 minutes, which is determined by the duration of the server's process launch.

Automatic adjustment

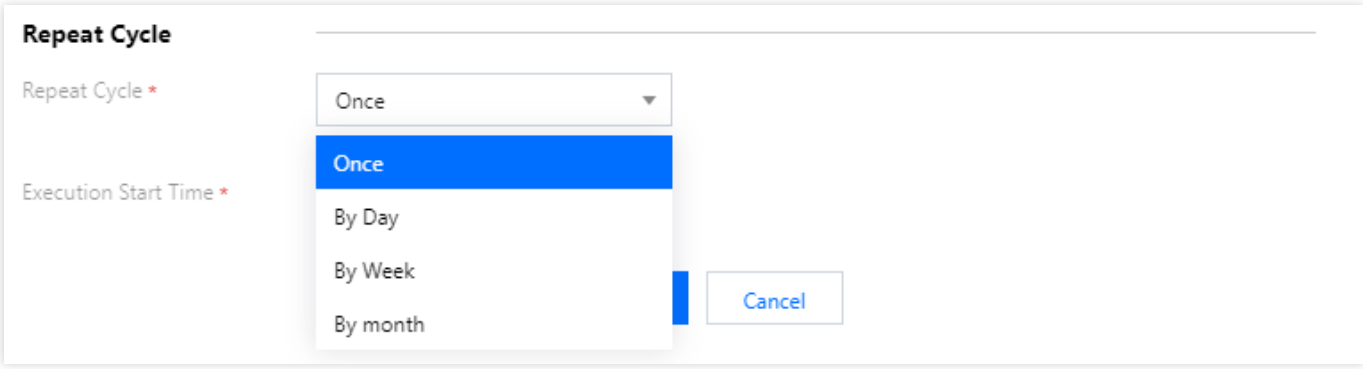


- **Instance Range**: it is in the format of minimum number of instances – maximum number of instances and is in integer type. Automatic adjustment cannot exceed the specified instance range. You cannot set this field if the first deployment of asset package fails.
- **Desired Quantity**: generally, 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 will adjust the number of instances with this metric.
- **Game Server Session Buffer**: represents the proportion of reserved idle instance resources for game server sessions, and automatic adjustment will scale with it.
- **Scaling Cooldown Time**: indicates the time interval between two scaling operations. It can be set from 1 to 30 minutes, which is determined by the duration of the server's process launch.

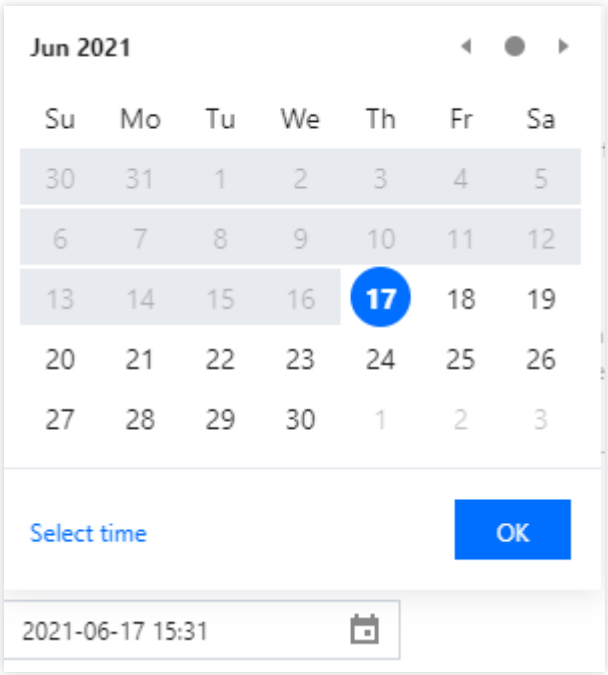
Note：
The sum of max instance range in a single region of an account cannot exceed the highest quota of each region. You can view the regional quotas in Resource Limits. To increase the CVM instance quota, please submit a ticket.

Repeat cycle

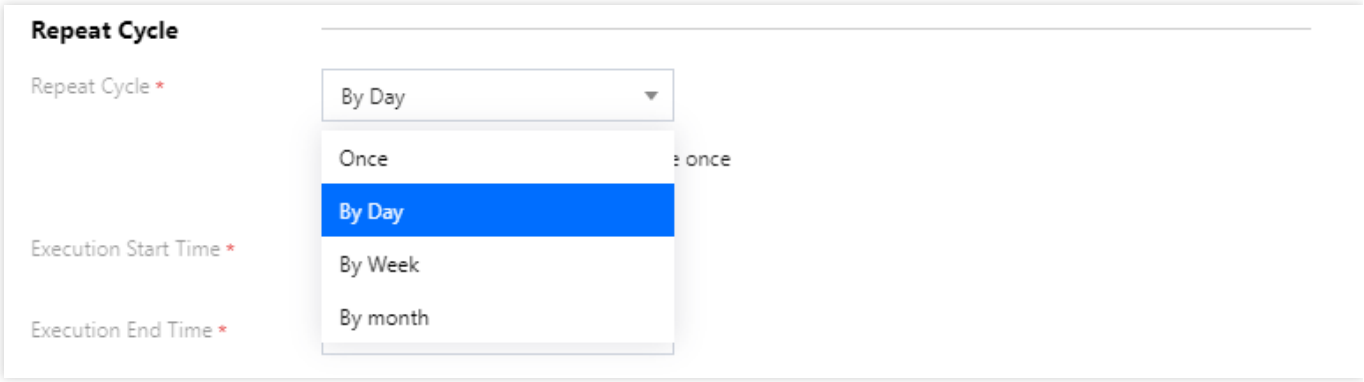
- **Repeat Cycle**: the scaling policy can be executed once, or can be repeated daily, weekly, and monthly.
- **Execution Start Time**: the start time of the scheduled update plan for scaling policy.
  i. If you select **Once** for the **Repeat Cycle**, the scaling policy will be executed once based on the specific time that you choose.



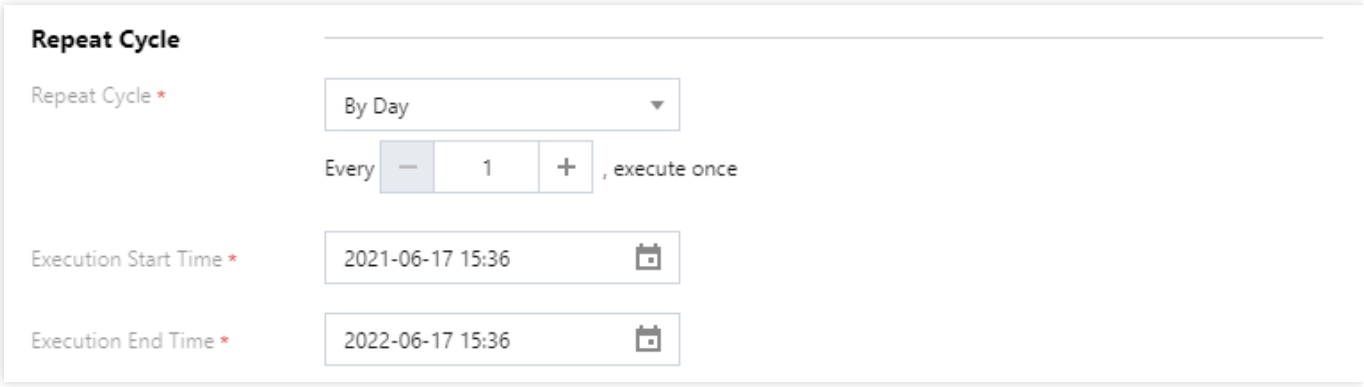
As shown in the figure below, the scaling policy will be executed once at 2021-06-17 15:31.



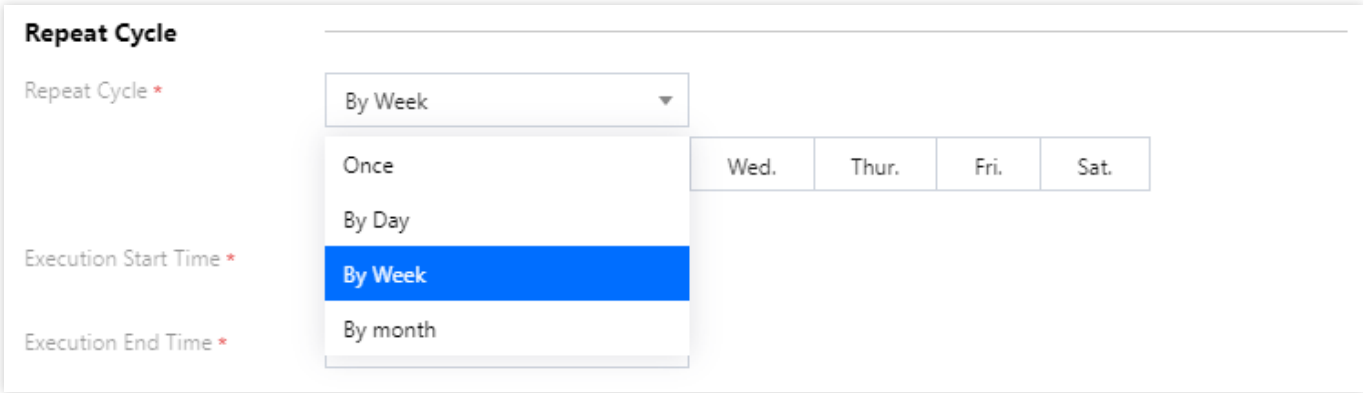
ii. If you select **By Day** for the **Repeat Cycle**, the scaling policy will be executed every N days based on the execution period that you choose.



As shown in the figure below, the scaling policy will be executed every other day from 2021-06-17 15:36 to 2022-06-17 15:36.

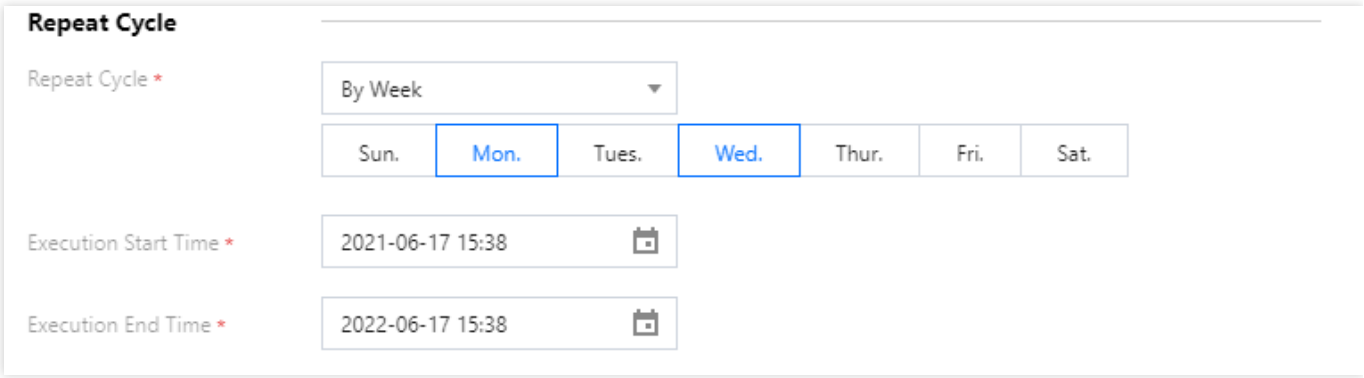


iii. If you select **By Week** for the **Repeat Cycle**, the scaling policy will be executed every week (based on the selected days) in the execution period that you choose.

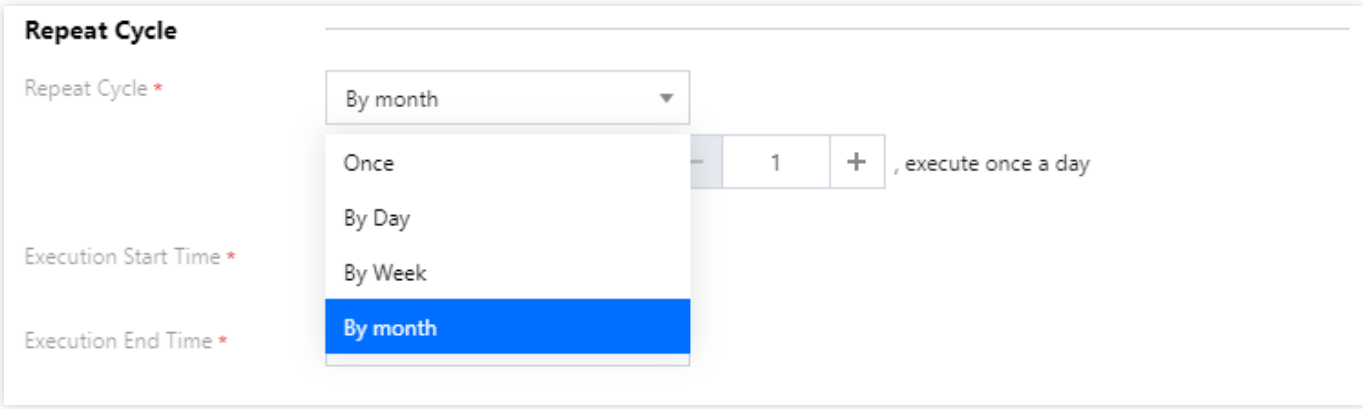


As shown in the figure below, the scaling policy will be executed every Monday and Wednesday

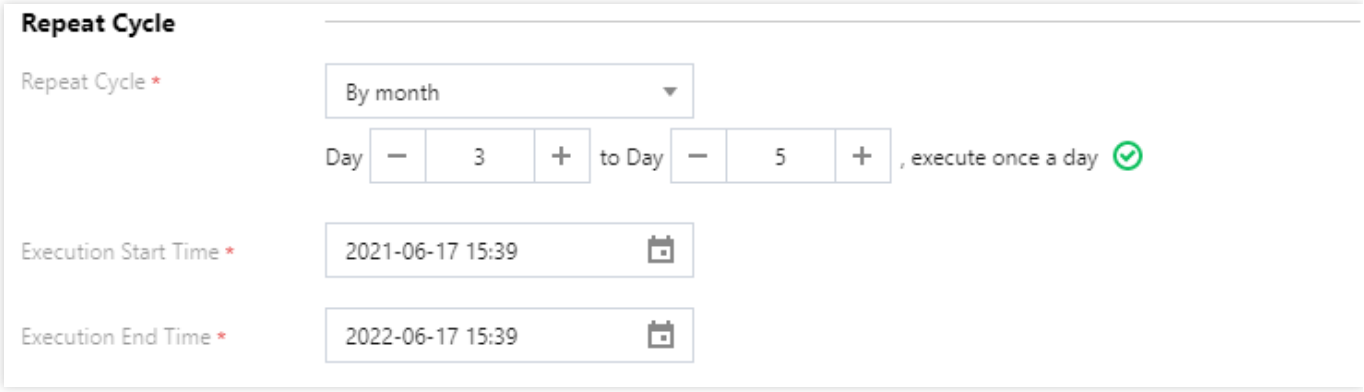
from 2021-06-17 15:38 to 2022-06-17 15:38.



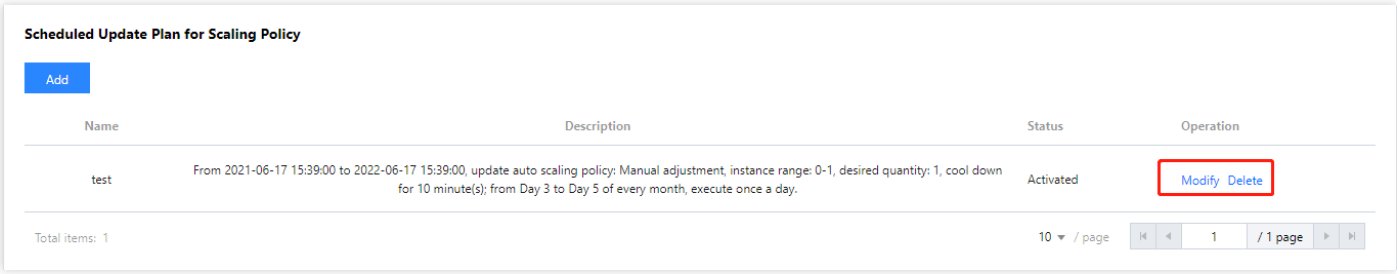
iv. If you select **By Month** for the **Repeat Cycle**, the scaling policy will be executed on the day N to the day M of each month in the execution period that you choose.



As shown in the figure below, the scaling policy will be executed on the 3rd to 5th day of each month from 2021-06-17 15:39 to 2022-06-17 15:39.



4. Click **Confirm**. After creating the policy, you can view, modify or delete it or perform other operations in the **Server Fleet Details** > **Scaling** page.



Viewing Game Server Session

Last updated : 2021-03-30 10:37:05

Overview

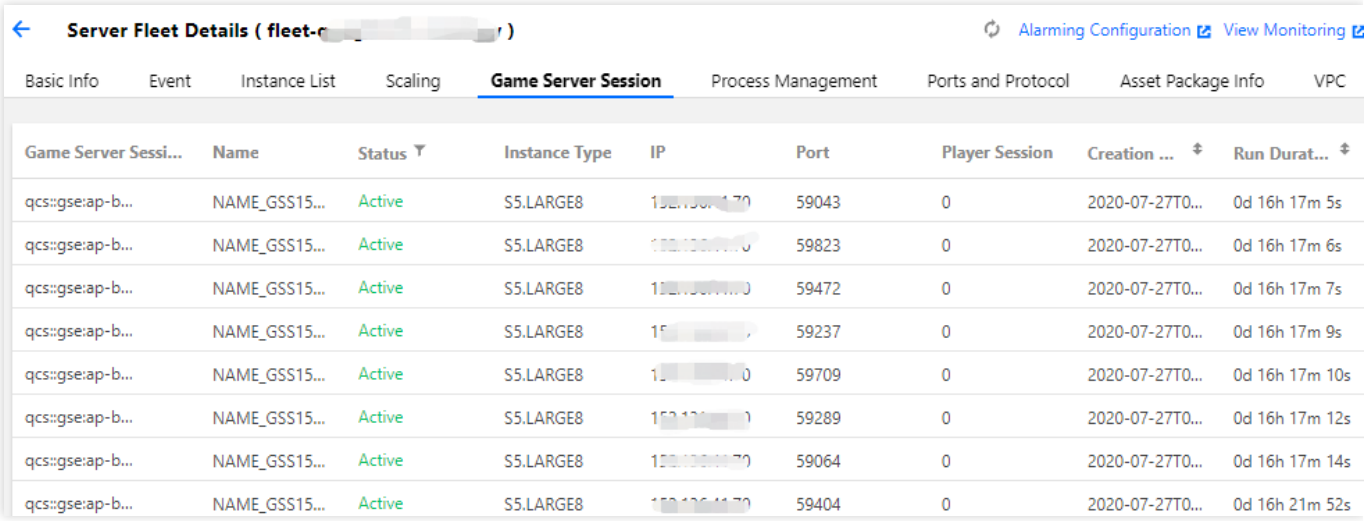
This document describes how to view a game server session, which corresponds to a server process. You can call a TencentCloud API to assign a game server session to a client, and GSE will assign this session to an idle process.

Prerequisites

You have created server fleets.

Directions

1. Log in to the GSE Console and create a server fleet. For more information, please see Creating Server Fleets.
2. Click the **ID** of the created server fleet to enter the fleet details page. Click the **Game Server Session** tab to view details as below:
   - **Game Server Session ID**: it is automatically generated by the system by default.
   - **Name**: it is entered when you call the CreateGameServerSession API.
   - **Status**: game server session statuses include undefined, active, activating, terminated, terminating, and abnormal.
   - **Instance Type**: it is automatically generated by the system by default.
   - **IP**: it is automatically generated by the system by default.
   - **Port**: it can be automatically generated by the system or assigned by you on the battle server.
   - **Player Sessions**: it indicates the number of player sessions in ACTIVE or RESERVED status.
     - ACTIVE sessions are sessions connected to the game server.
     - RESERVED sessions are sessions where players have been assigned with slots in the game server session but have not connected.
   - **Creation Time**: it indicates the creation time of game server session in the format of `yyyy-mm-dd hh:mm:ss` . The entries can be sorted in ascending/descending order.
   - **Run Duration**: it indicates the duration between running start time and current time in the format of `xxd xxh xxm xxs` . The entries can be sorted in ascending/descending order.



Associating CCN Instance

Last updated : 2022-01-21 10:53:22

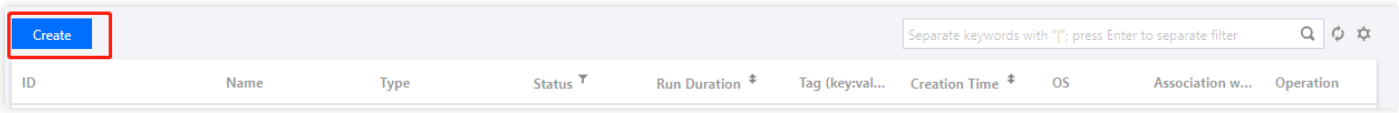
Overview

This document describes how to associate CCN to enable multi-point private interconnection, implementing multi-point interconnection in all regions on and off the cloud.

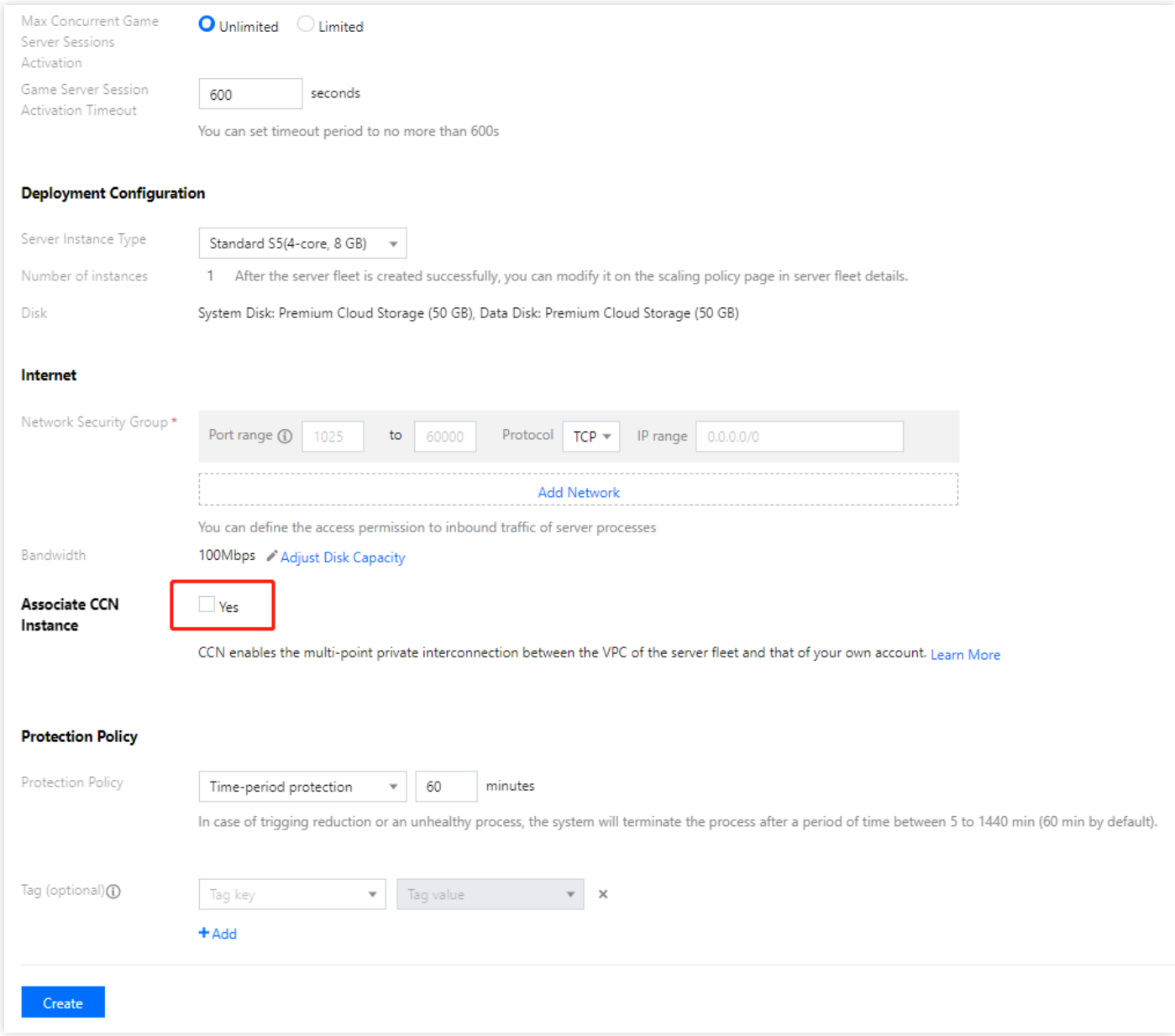
Directions

Associating with a CCN instance during the creation of the server fleet

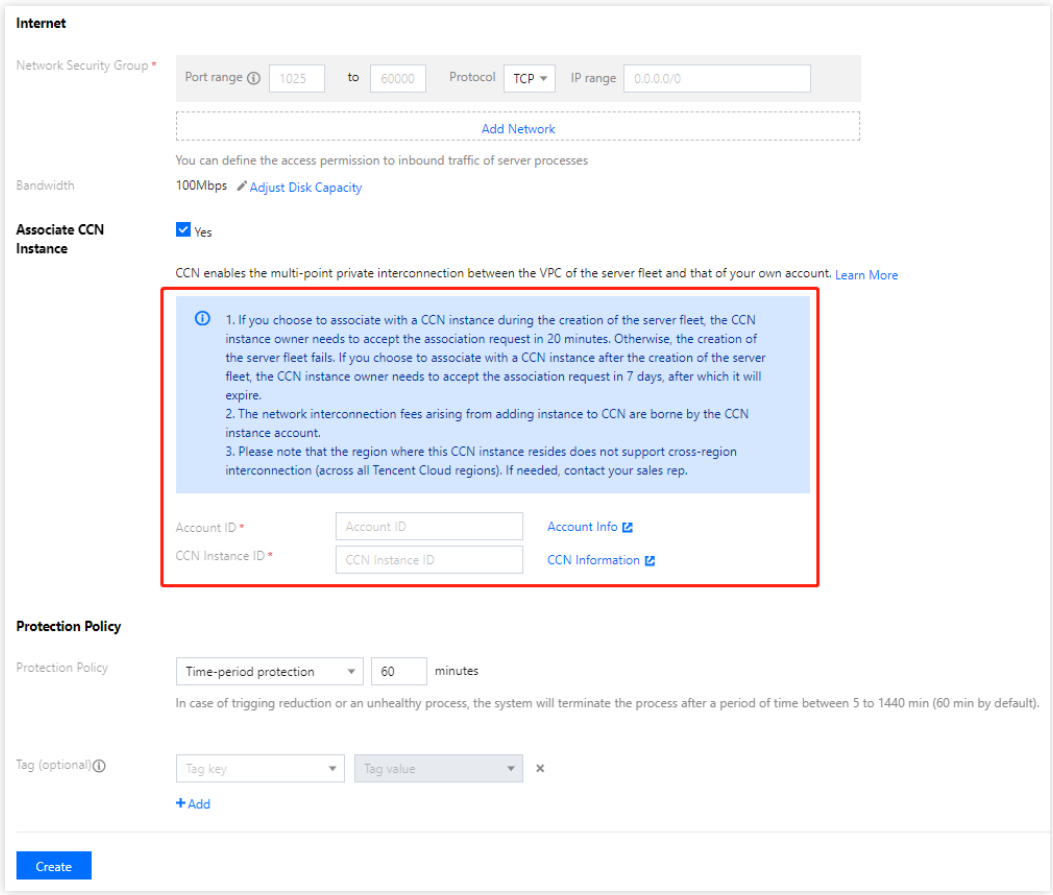
1. Log in to the GSE console and click **Fleet** on the left sidebar to enter the server fleet page.
2. Click **Create** to create a server fleet.



3. Select **Associate CCN Instance** on the **Create Fleet** page.



4. Enter the **Account ID** and **CCN Instance ID** for association.



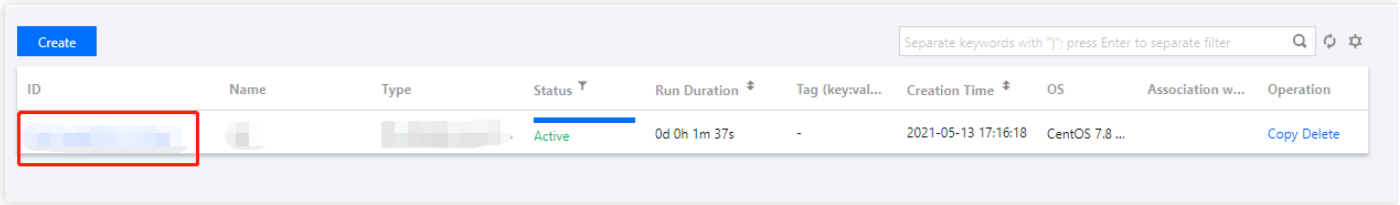
Note :

1. If you choose to associate with a CCN instance during the creation of the server fleet, the CCN instance owner needs to accept the association request in 20 minutes. Otherwise, the creation of the server fleet fails.
2. The network interconnection fees arising from adding instance to CCN are borne by the CCN instance account.
3. Please note that the region where this CCN instance resides does not support cross-region interconnection (across all Tencent Cloud regions). If needed, contact your sales rep.

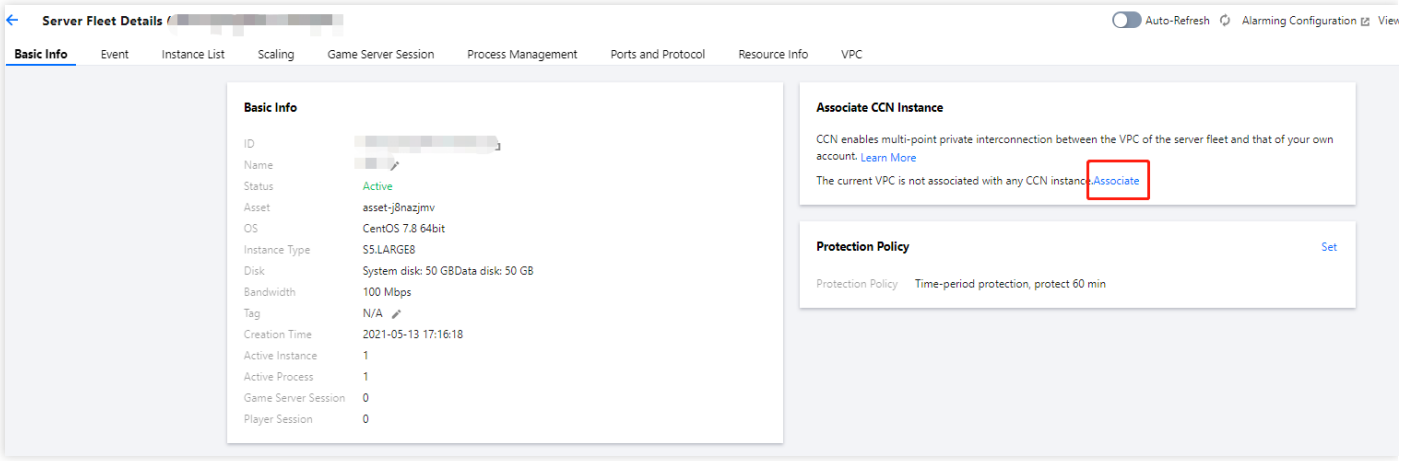
Associating with a CCN instance after the creation of the server fleet

1. Log in to the GSE console and click **Fleet** on the left sidebar to enter the server fleet page.

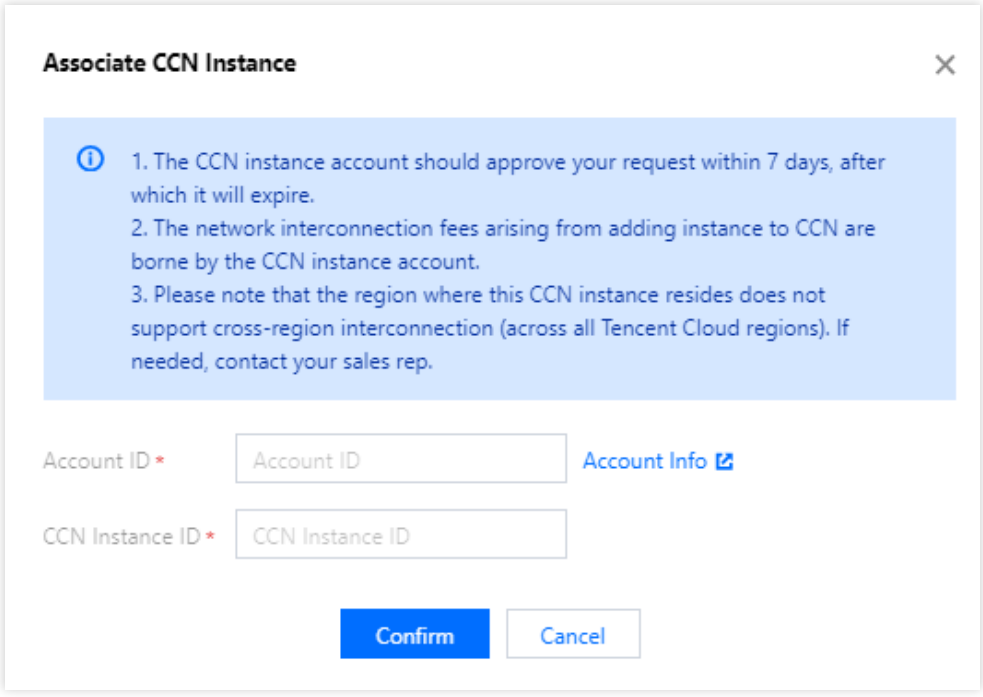
2. Created a server fleet.
3. Click the **ID** of the server fleet that needs to associate with a CCN instance to go to its details page.



4. On the details page, click **Associate** in the **Associate CCN Instance** section.



5. Enter the **Account ID** and **CCN Instance ID** for association.



Note：

1. If you choose to associate with a CCN instance after the creation of the server fleet, the CCN instance owner needs to accept the association request in 7 days, after which it will expire.
2. The network interconnection fees arising from adding instance to CCN are borne by the CCN instance account.
3. Please note that the region where this CCN instance resides does not support cross-region interconnection (across all Tencent Cloud regions). If needed, contact your sales rep.

Creating Aliases

Last updated : 2021-03-30 10:25:31

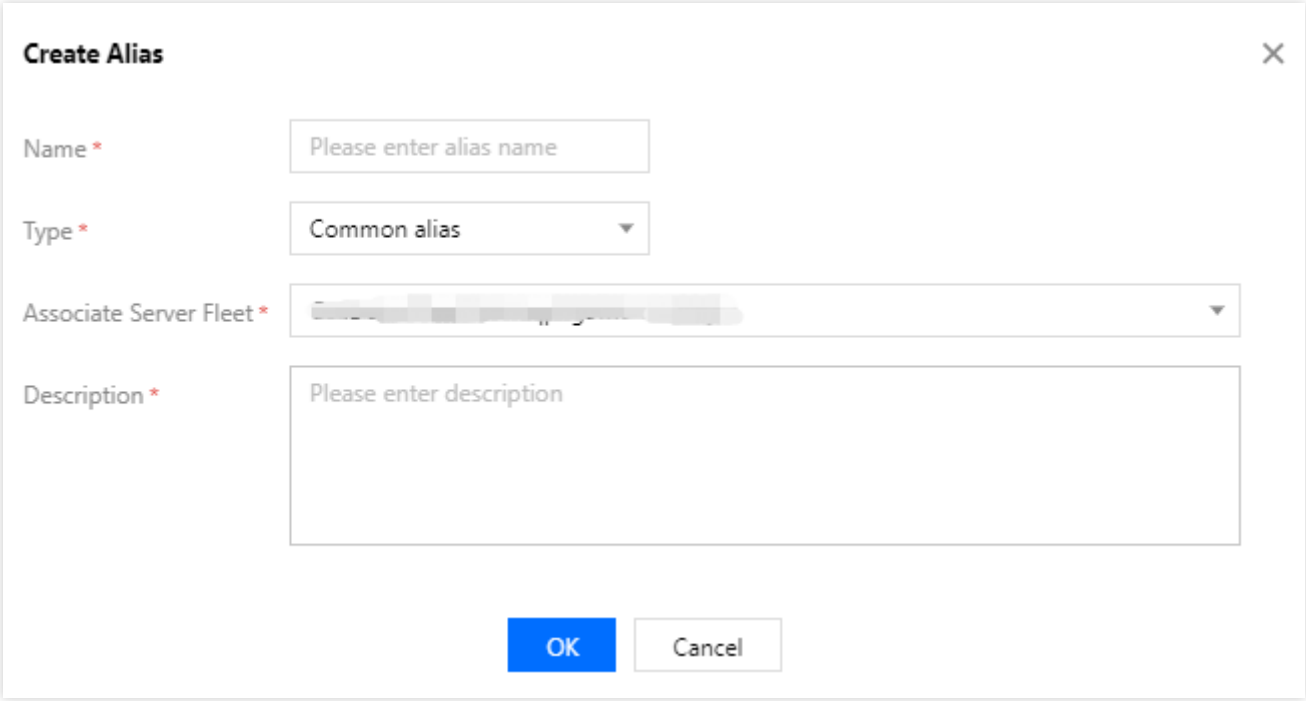
Overview

This document describes how to create an alias to abstract the name of a fleet. By using an alias instead of a specific fleet ID, you can change the server fleet associated with the alias to switch player traffic from one fleet to another fleet more easily and seamlessly for zero downtime updates.

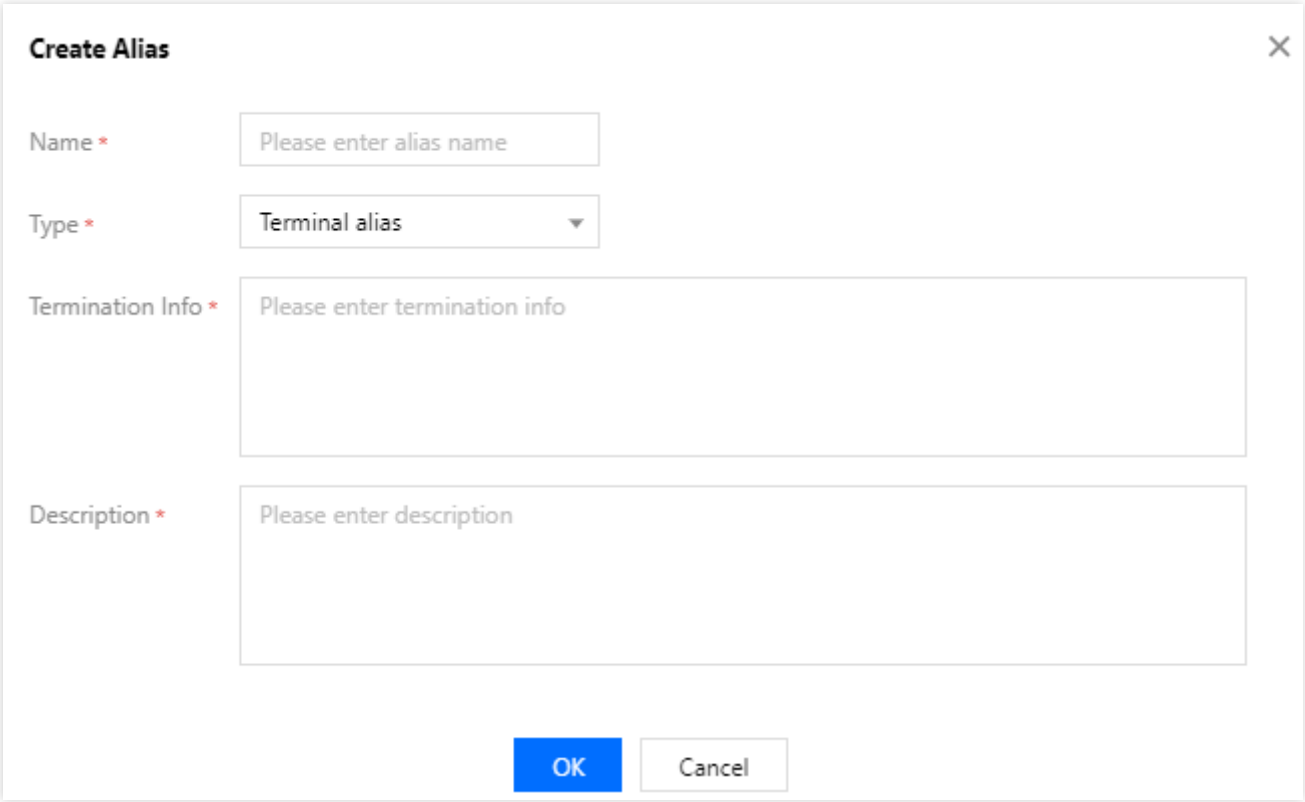
Directions

1. Log in to the GSE Console and click **Alias** on the left sidebar.
2. Click **Create** in the top-left corner.
3. In the alias creation page, enter information such as the name, type, and description, and click **OK**.
   - **Name**: enter the alias name for easy identification in the directory.
   - **Type**: select the alias type in the drop-down list. The types include common alias and terminal alias.
     - 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**: select the server fleet to be associated with if "Type" is "Common alias".
- **Termination Info**: enter the information to be displayed to players if "Type" is "Terminal alias".
- **Description**: enter a short description of the alias for easy identification.

4. After the alias is successfully created, you can modify or delete it or perform other operations on it on the alias page.

Create Game Server Queue

Last updated : 2021-05-20 10:10:24

Overview

This document describes how to create a game server queue. You can use a queue to create a group of cross-region server fleets and place game server sessions in any fleet in the queue, so as to minimize latency, deliver a better player experience, use more fleet capacity more efficiently, provide high capacity for new games more swiftly, and make the game more elastically available.

Prerequisites

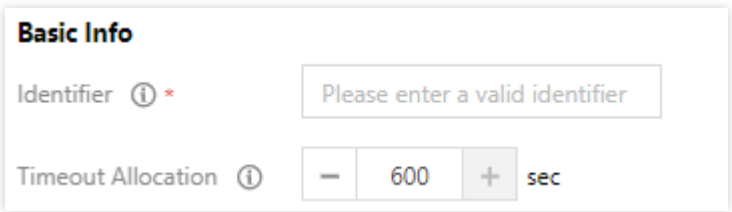
You have created server fleets.

Directions

1. Log in to the GSE Console and click **Queue** on the left sidebar.
2. Click **Create** in the top-left corner.
3. Enter information such as Basic Info, Latency Policy, and Target.

Basic info

- Identifier: enter a valid identifier.
- Timeout Allocation: enter the max time that a game server session request can be retained in a multi-region deployment.

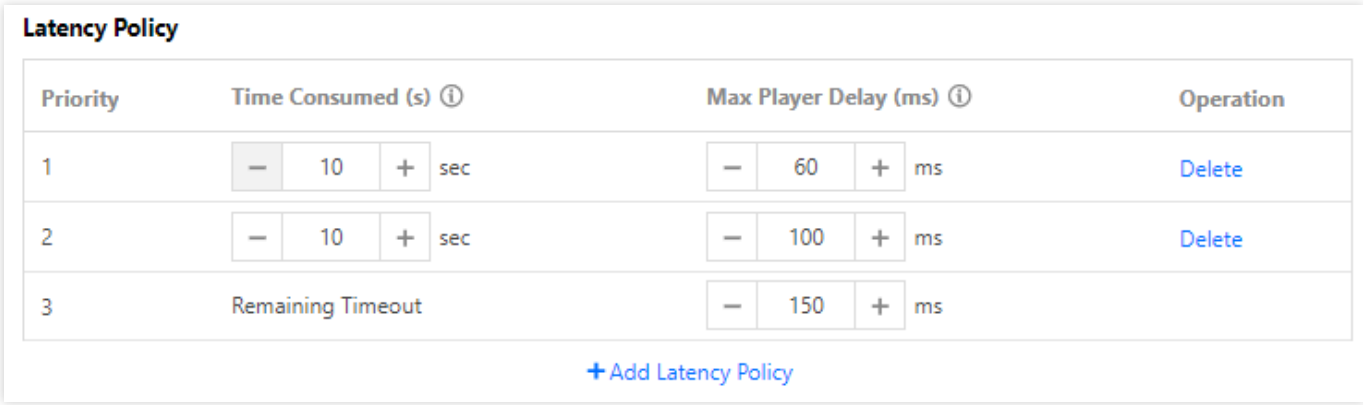


Latency policy

Set a group of policies for max player latency which increases as the policy priority is lowered. You can use the policies that are executed in priority order to strike a balance between the optimal game

experience and the player wait time.

- **Example**:
  Define the following policies for a queue whose timeout duration is 30 seconds. Specify the max latency as 60 ms for the first policy, and then keep specifying a greater value for the next policy until the last one (150 ms) which defines the absolute max latency for any player. To ensure all game server sessions are placed regardless of the latency, you can set the max latency to a very high value for the last policy.
  - **Policy settings**:



  - **Execution result**:
    In this example, the first policy uses the first 10 seconds of the timeout duration to search for a fleet whose latency for any player is below 60 milliseconds. The second policy uses the next 10 seconds to search for a fleet whose latency for any player is below 100 milliseconds. The third policy uses the last 10 seconds to search for a fleet whose latency for any player is below 150 milliseconds. If there are no fleets whose player latency is below the max player latency in a policy, GSE will return the result directly without waiting until the specified timeout in the policy. If there is a fleet whose latency is below the max player latency, but it is not idle, GSE will wait until it times out.

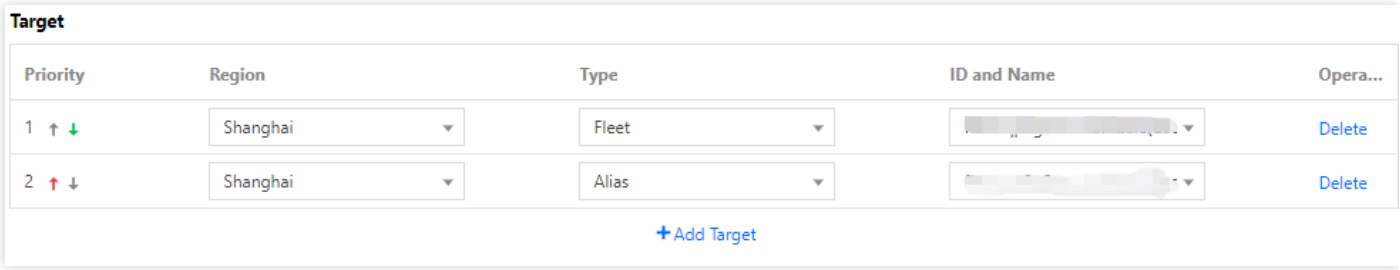
Target

The target is a list of server fleets where GSE schedules resources. You can specify a fleet with a server fleet ID or alias ID and set the priority for the targets in the queue. If you set both latency policy and target, the latency policy shall prevail. When selecting targets for a queue, please consider the following points:

- **The priority of targets in the queue really matters.** If you provide latency data in a game server session placement request, GSE will adjust the target priority to find the available resources with the minimum player latency; otherwise, GSE will follow the priority order in the target list,

and in this case, game sessions will generally be hosted on the first listed fleet and will be placed on a backup fleet only when necessary.

- **You can add any existing server fleets or aliases from any region.**
- **A queue should contain at least two fleets, which should be in at least two different regions if possible,** as this can reduce the impact of regional speed-down of fleets, manage unexpected changes in player needs more efficiently, and make service hosting more elastic.
- **If you have associated a server fleet with an alias (recommended), you'd better use the alias when setting the fleet as a target in the queue.**
- **All targets in the queue must run a game asset package compatible with the game clients of this queue.** Please note that new game session requests processed by the queue may be placed on any target in the queue.
- **You need to determine where in the region your queue is to be created.** Ideally, you can initiate game server session placement requests through a game client service (such as session directory service). To implement this, we recommend you create a queue in a region close to where the client service is deployed, which can minimize the latency in submitting such requests.



4. Click **OK** to create a game server queue. Once created successfully, you can modify or delete it or perform other operations on the queue list page.

Viewing Monitoring Information

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

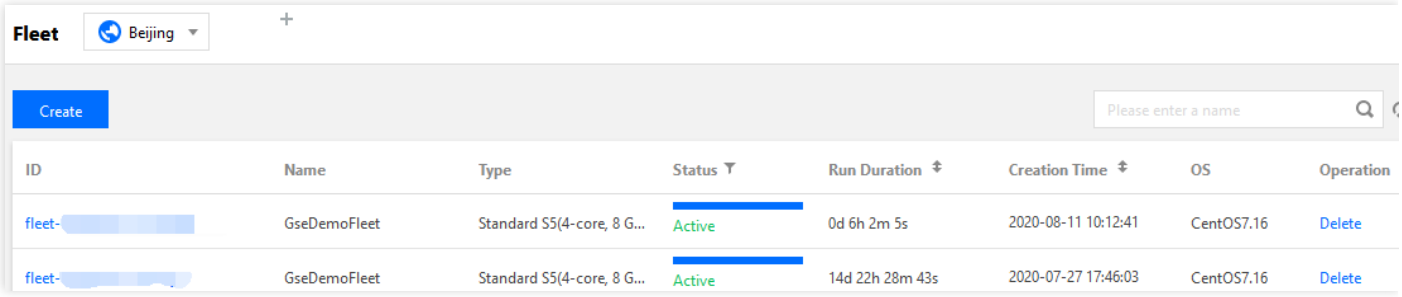
This document describes how to view monitoring information such as server fleets, game server queues, and instances.

Prerequisites

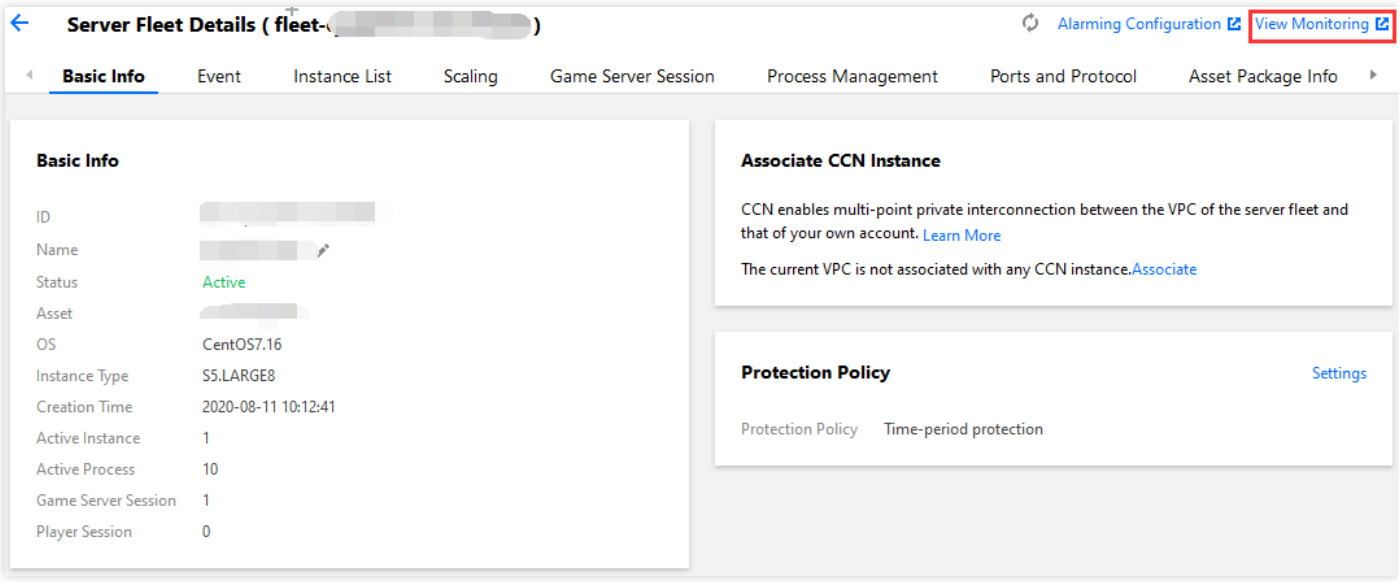
You have created a server fleet.

Directions

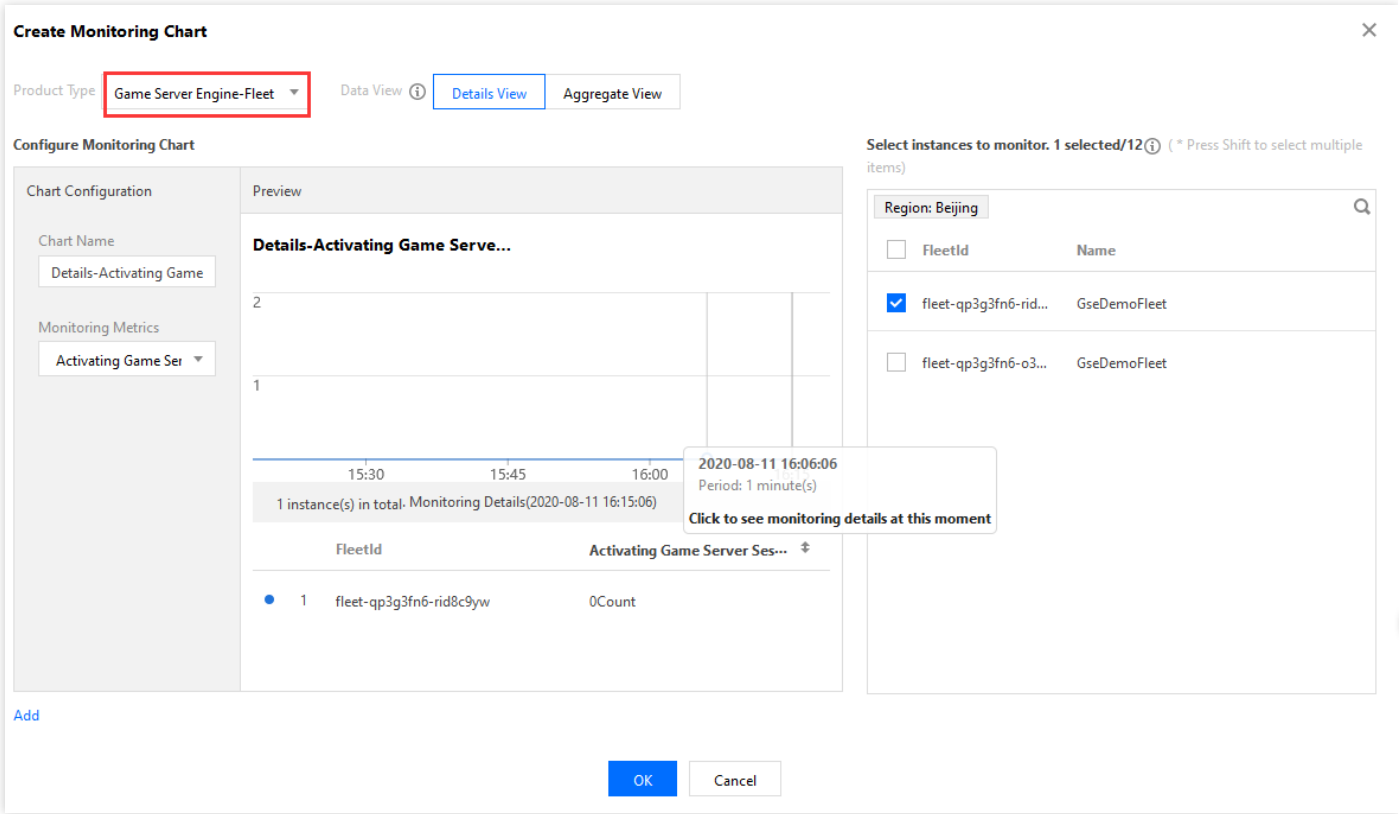
1. Log in to the GSE console and click **Fleet** on the left sidebar.
2. Click the server fleet ID to enter the server fleet details page.



3. On the server fleet details page, click **View Monitoring** in the top-right corner to enter the monitoring panel.



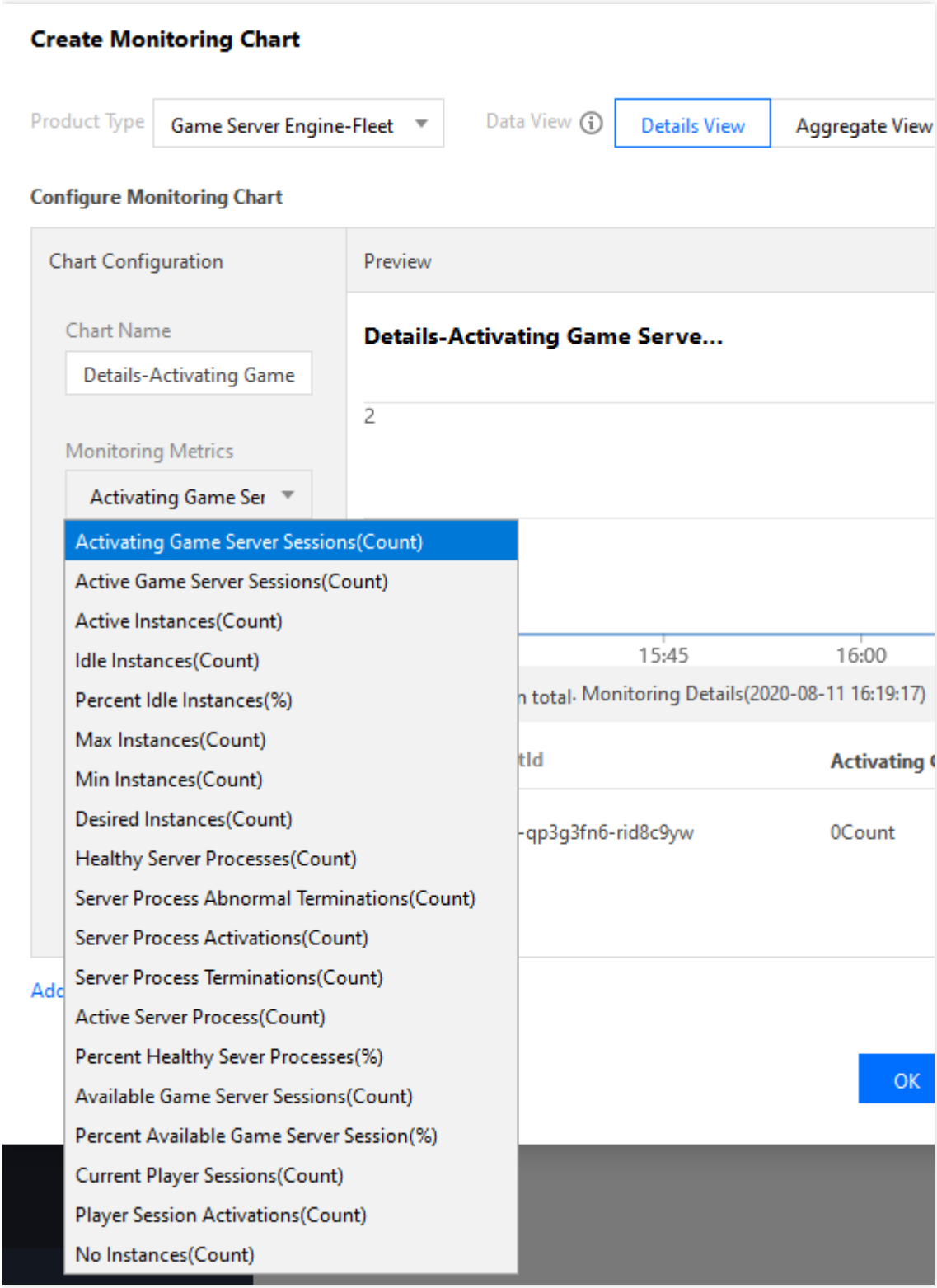
4. Click **Dashboard** > **Dashboard List** > **Create Dashboard** > **Create Chart** to create a monitoring chart.
5. Select the namespace and metrics in the **Metric** to define data to be displayed on the monitoring chart. You can select **Fleet**, **GameServerSessionQueue** or **Instance** as the namespace as needed.



- Description of monitoring metrics available for **Game Server Elastic-scaling - Fleet**:

Monitoring Metric	Description
Activating Game Server Sessions (Count)	Number of game server sessions in ACTIVATING status (a session in this status is being launched)
Active Game Server Sessions (Count)	Number of game server sessions in ACTIVE status (a session in this status can host players and is hosting zero to multiple players)
Active Instances (Count)	Number of instances in ACTIVE status (an instance in this status is running active server processes)
Idle Instances (Count)	Number of active instances that are not hosting any game server sessions
Percent Idle Instances (%)	Percentage of active instances in idle status
Max Instances (Count)	Maximum number of instances allowed by the server fleet
Min Instances (Count)	Minimum number of instances allowed by the server fleet
Desired Instances (Count)	Target number of active instances that can be maintained by the server fleet
Healthy Server Processes (Count)	Number of active server processes that are running normally
Server Process Abnormal Terminations (Count)	Number of server processes that have been closed due to exceptions since the last report
Server Process Activations (Count)	Number of server processes that have successfully switched from the ACTIVATING status to ACTIVE status since the last report
Server Process Terminations (Count)	Number of server processes that have been closed since the last report
Active Server Process (Count)	Number of server processes in ACTIVE status (a process in this status is running and can host game server sessions)
Percent Healthy Server Processes (%)	Percentage of all active server processes that are running normally
Available Game Server Sessions (Count)	Number of idle game server session slots in the active server processes that are running normally
Percent Available Game Server Session (%)	Percentage of idle game server session slots in all active server processes (no matter whether they are running normally or not)

Monitoring Metric	Description
Active Player Sessions (Count)	Player sessions in ACTIVE (the players have connected to active game server sessions) or RESERVED status (the players have been assigned with slots in the game server session but have not connected)
Player Session Activations (Count)	Number of player sessions that have switched from the RESERVED status to ACTIVE status since the last report (within a certain period of time)
No Instances (Count)	Number of instances that fail to be purchased

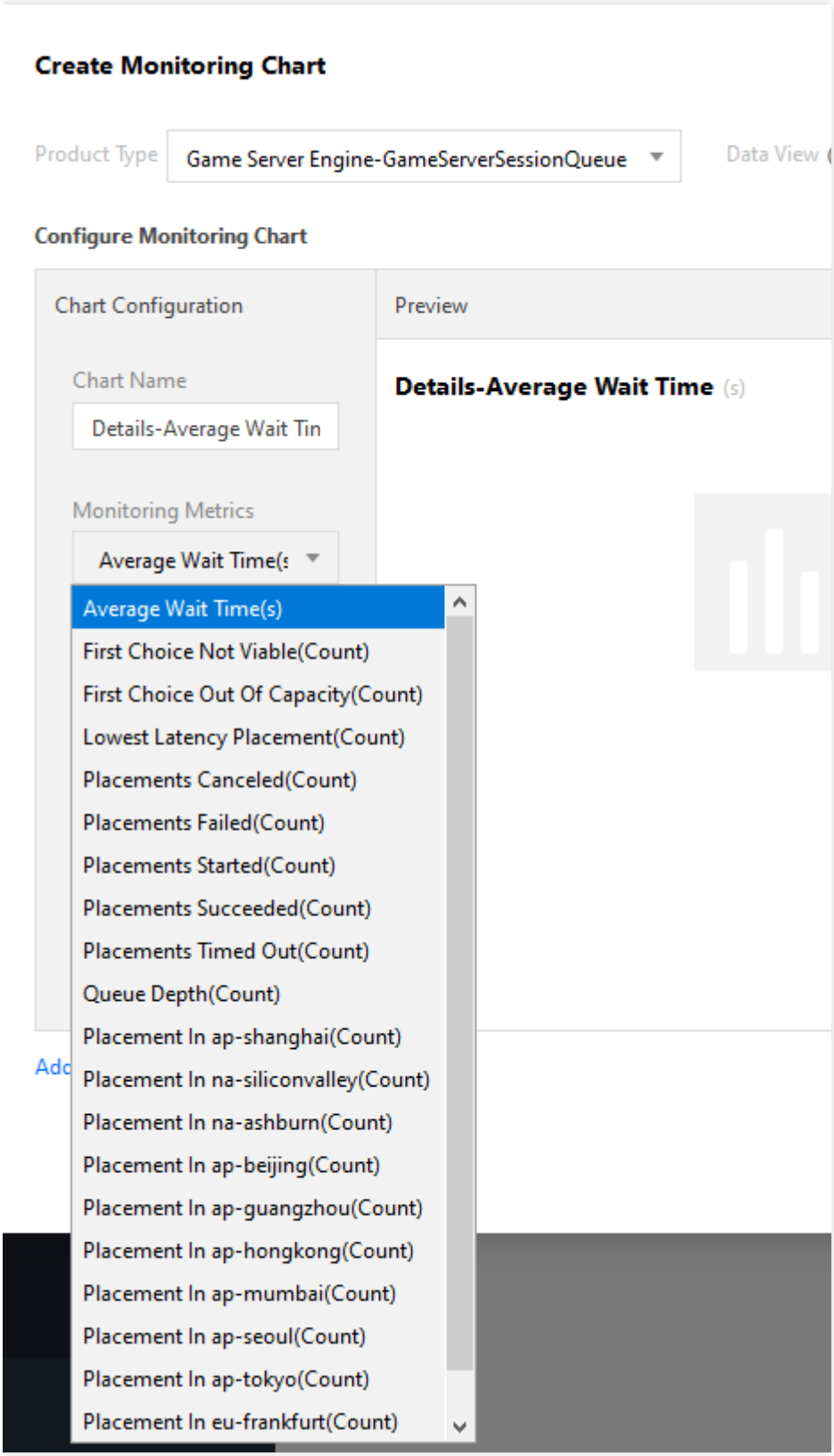


- Description of available monitoring metrics for **Game Server Elastic-scaling - GameServerSessionQueue**:

Monitoring Metric	Description

Monitoring Metric	Description
Average Wait Time (s)	Average execution wait time for game server session placement requests in PENDING status in game server queue
First Choice Not Viable (Count)	Number of game server sessions that are placed in to a fleet other than the preferred one as the preferred one is nonviable (for example, it is a Spot fleet with a high interruption rate)
First Choice Out of Capacity (Count)	Number of game server sessions that are placed in to a fleet other than the preferred one as the preferred fleet does not have available resources
Lowest Latency Placement (Count)	Number of game server sessions that are placed into the region with the lowest latency for players
Placements Canceled (Count)	Number of game server session placement requests that have been canceled before timeout since the last report
Placements Failed (Count)	Number of failed game server session placement requests since the last report
Placements Started (Count)	Number of new game server session placement requests that have been added to the queue since the last report
Placements Succeeded (Count)	Number of successful game server session placement requests since the last report
Placements Timed Out (Count)	Number of game server session placement requests that are not executed due to the queue timeout since the last report
Queue Depth (Count)	Number of game server session placement requests in PENDING status in queue
Placement In ap-shanghai (Count)	Number of game server sessions that have been successfully placed in a server fleet in Shanghai since the last report
Placement In na-siliconvalley (Count)	Number of game server sessions that have been successfully placed in a server fleet in Silicon Valley since the last report
Placement In na-ashburn (Count)	Number of game server sessions that have been successfully placed in a server fleet in Virginia since the last report
Placement In ap-beijing (Count)	Number of game server sessions that have been successfully placed in a server fleet in Beijing since the last report

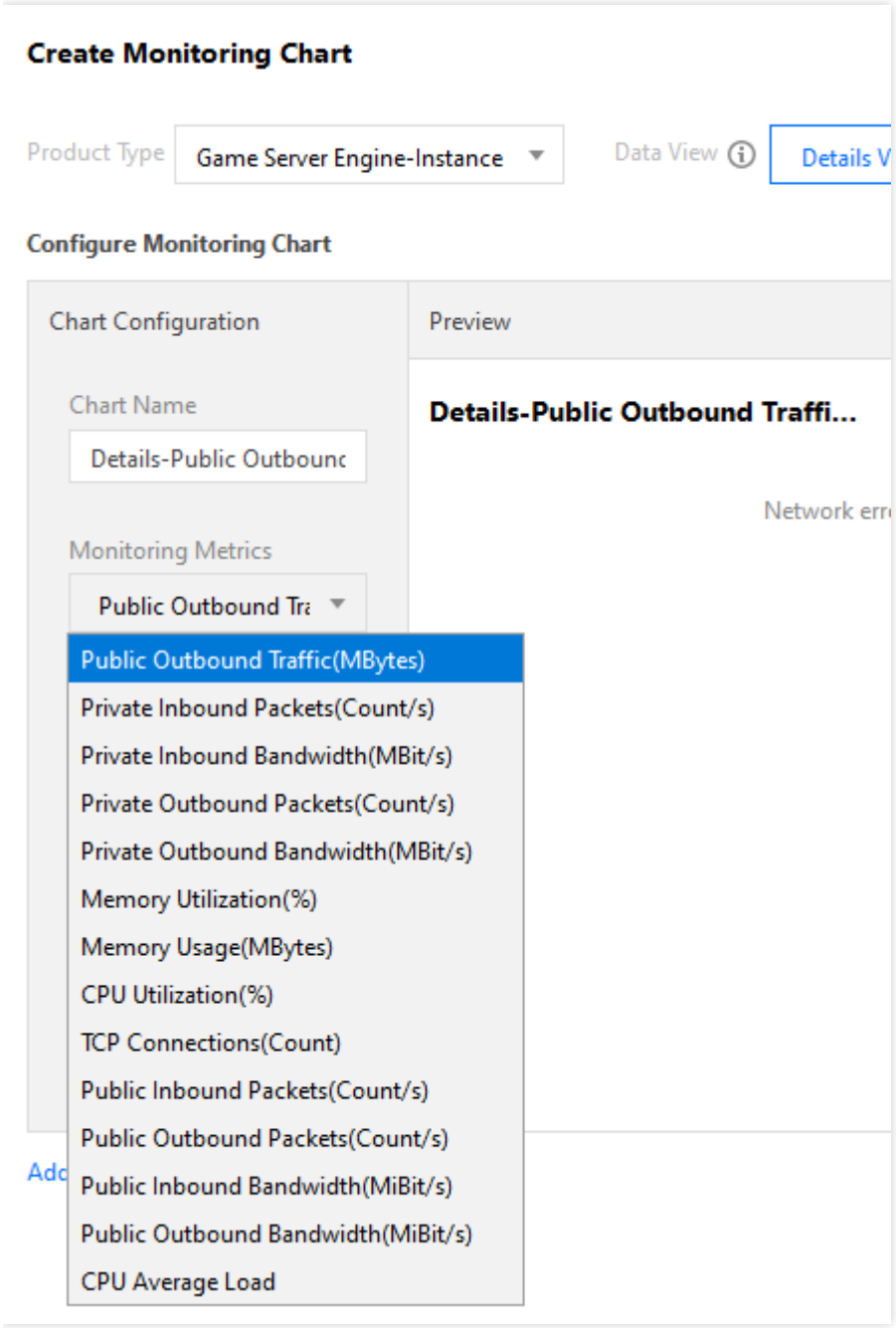
Monitoring Metric	Description
Placement In ap-guangzhou (Count)	Number of game server sessions that have been successfully placed in a server fleet in Guangzhou since the last report
Placement In ap-hongkong (Count)	Number of game server sessions that have been successfully placed in a server fleet in Hong Kong since the last report
Placement In ap-mumbai (Count)	Number of game server sessions that have been successfully placed in a server fleet in Mumbai since the last report
Placement In ap-seoul (Count)	Number of game server sessions that have been successfully placed in a server fleet in Seoul since the last report
Placement In ap-tokyo (Count)	Number of game server sessions that have been successfully placed in a server fleet in Tokyo since the last report
Placement In eu-frankfurt (Count)	Number of game server sessions that have been successfully placed in a server fleet in Frankfurt since the last report
Placement In ap-singapore (Count)	Number of game server sessions that have been successfully placed in a server fleet in Singapore since the last report
Placement In ap-bangkok (Count)	Number of game server sessions that have been successfully placed in a server fleet in Bangkok since the last report



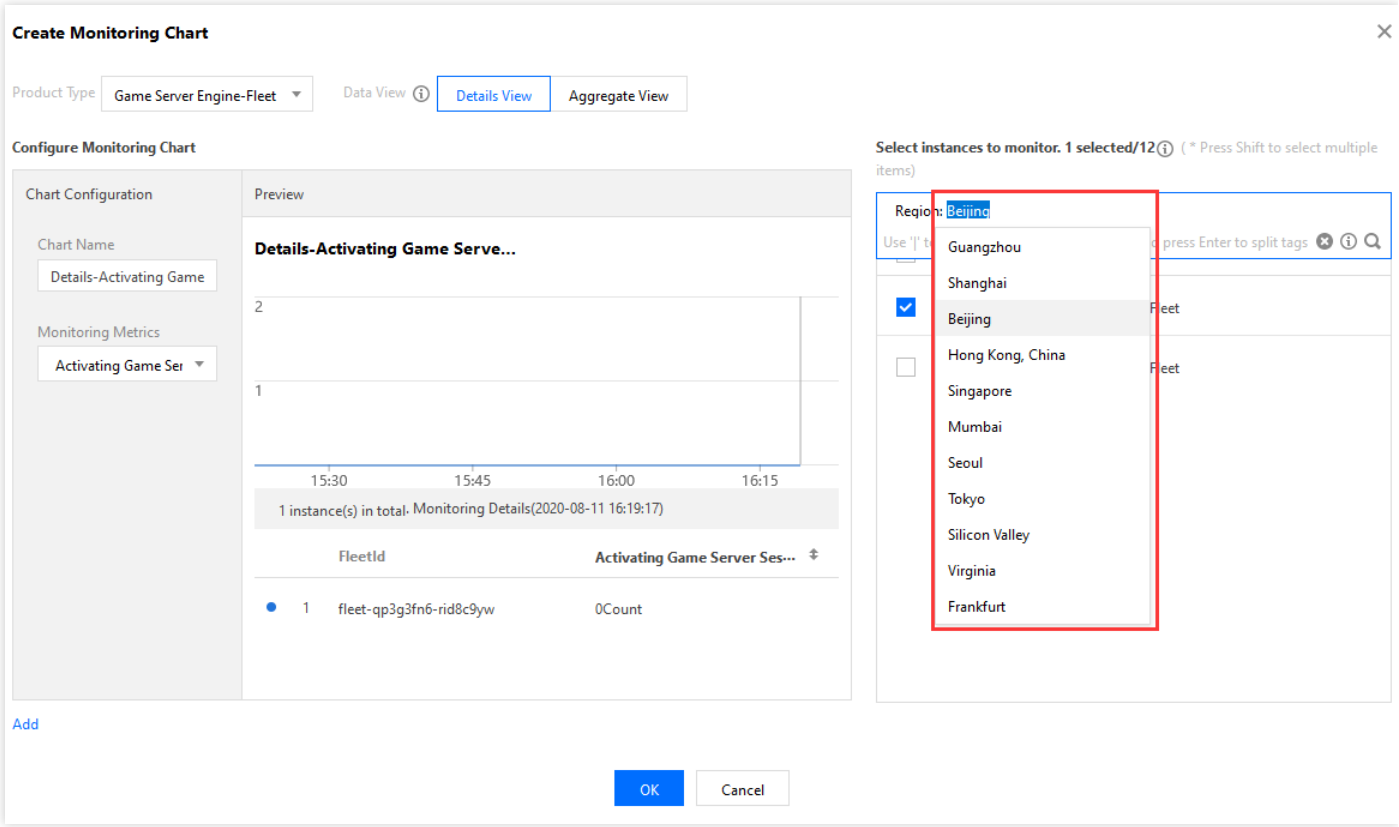
- Description of available monitoring metrics for **Game Server Elastic-scaling - Instance**:

Monitoring Metric	Description
Public Outbound Traffic (MBytes)	Average outbound traffic per second of public ENI
Private Inbound Packets (Count/s)	Average number of inbound packets per second of private ENI
Private Inbound Bandwidth (MBit/s)	Average inbound traffic per second of private ENI
Private Outbound Packets (Count/s)	Average number of outbound packets per second of private ENI
Private Outbound Packets (Count/s)	Average outbound traffic per second of private ENI
Memory Utilization (%)	Percentage of the memory actually used, excluding the memory used by buffers and system caches
Memory Usage (MBytes)	Amount of the memory actually used, excluding the memory used by buffers and system caches
CPU Utilization	Real time CPU utilization during instance execution
TCP Connections (Count)	Number of TCP connections in ESTABLISHED status
Public Inbound Packets (Count/s)	Average number of inbound packets per second of public ENI
Public Outbound Packets (Count/s)	Average number of outbound packets per second of public ENI

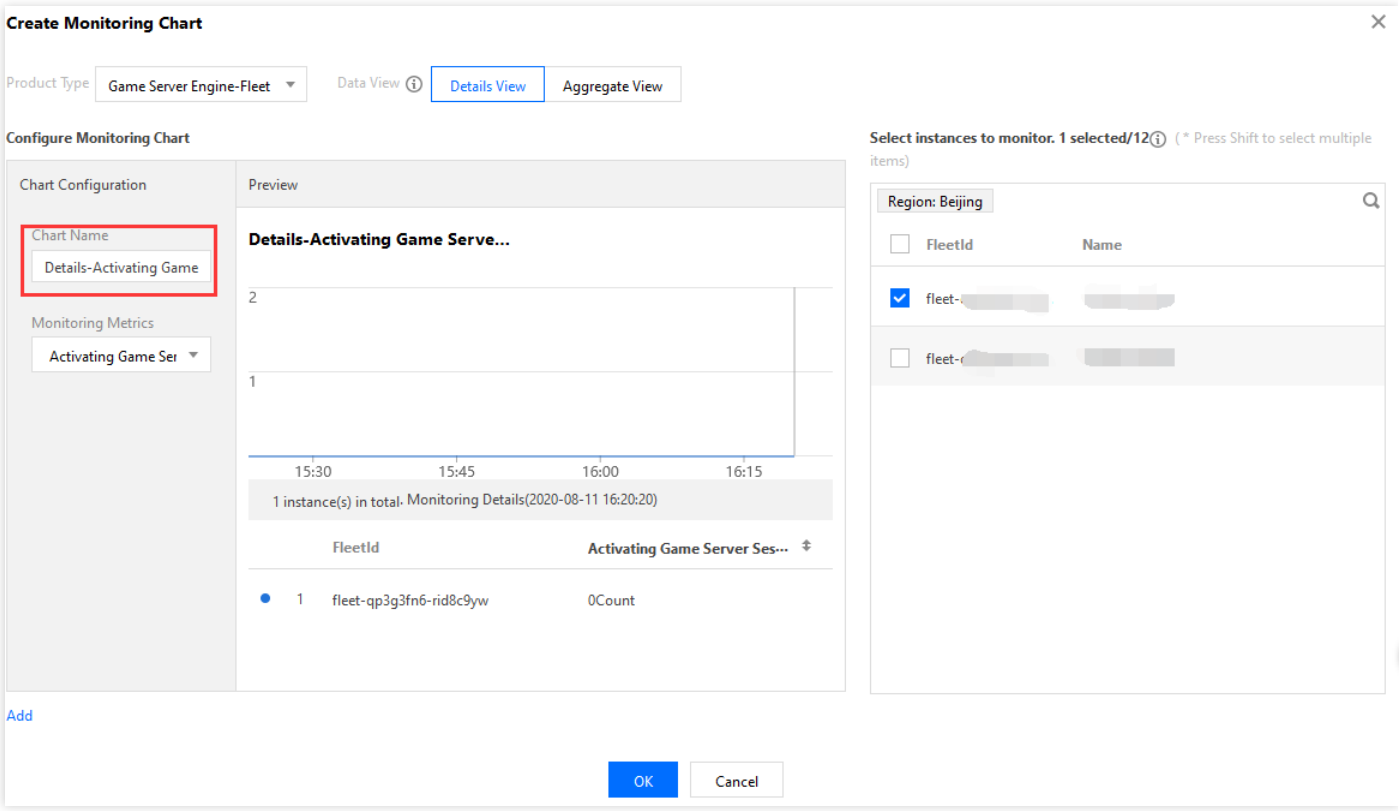
Monitoring Metric	Description
Public Inbound Bandwidth (MiBit/s)	Average inbound traffic per second of public ENI
Public Outbound Bandwidth (MiBit/s)	Average outbound traffic per second of public ENI
CPU Average Load	Average number of tasks that are using and waiting to use the CPU per minute



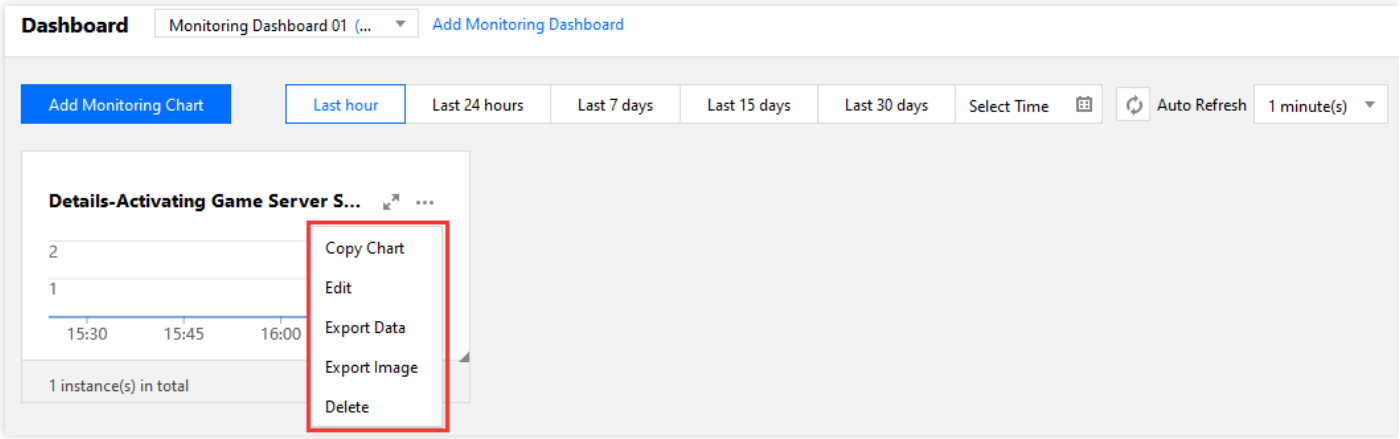
6. Select the region of the object to be monitored in **Region** on the right, and the list of monitoring object names will be displayed for you to select.



7. You can click **Chart Name** to rename the chart and click **OK** to create a monitoring chart as needed.



8. You can also copy, edit, export and delete the chart later when needed.



Viewing Statistics

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

This document describes how to view the total CVM usage period, network traffic and other information of server fleets in different regions through the statistics feature.

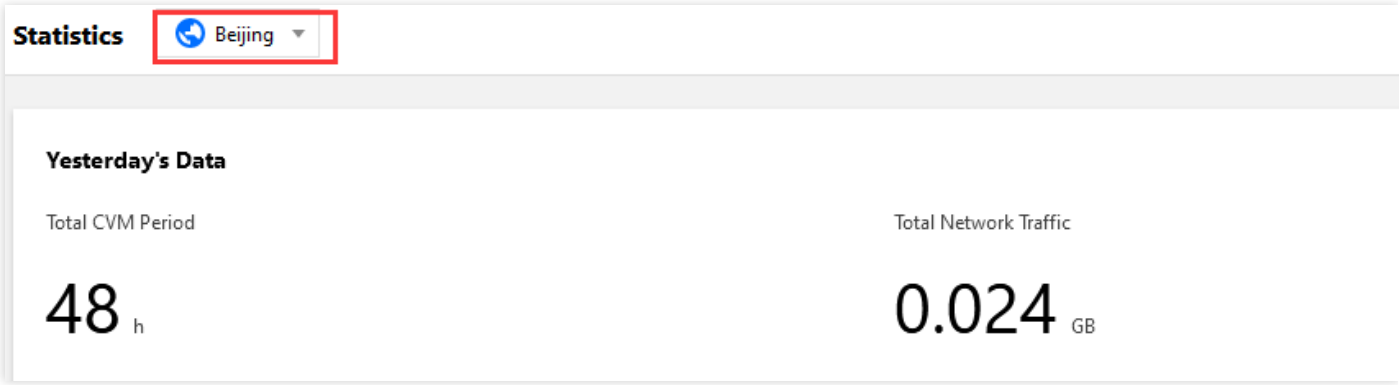
Prerequisite

You have created a server fleet.

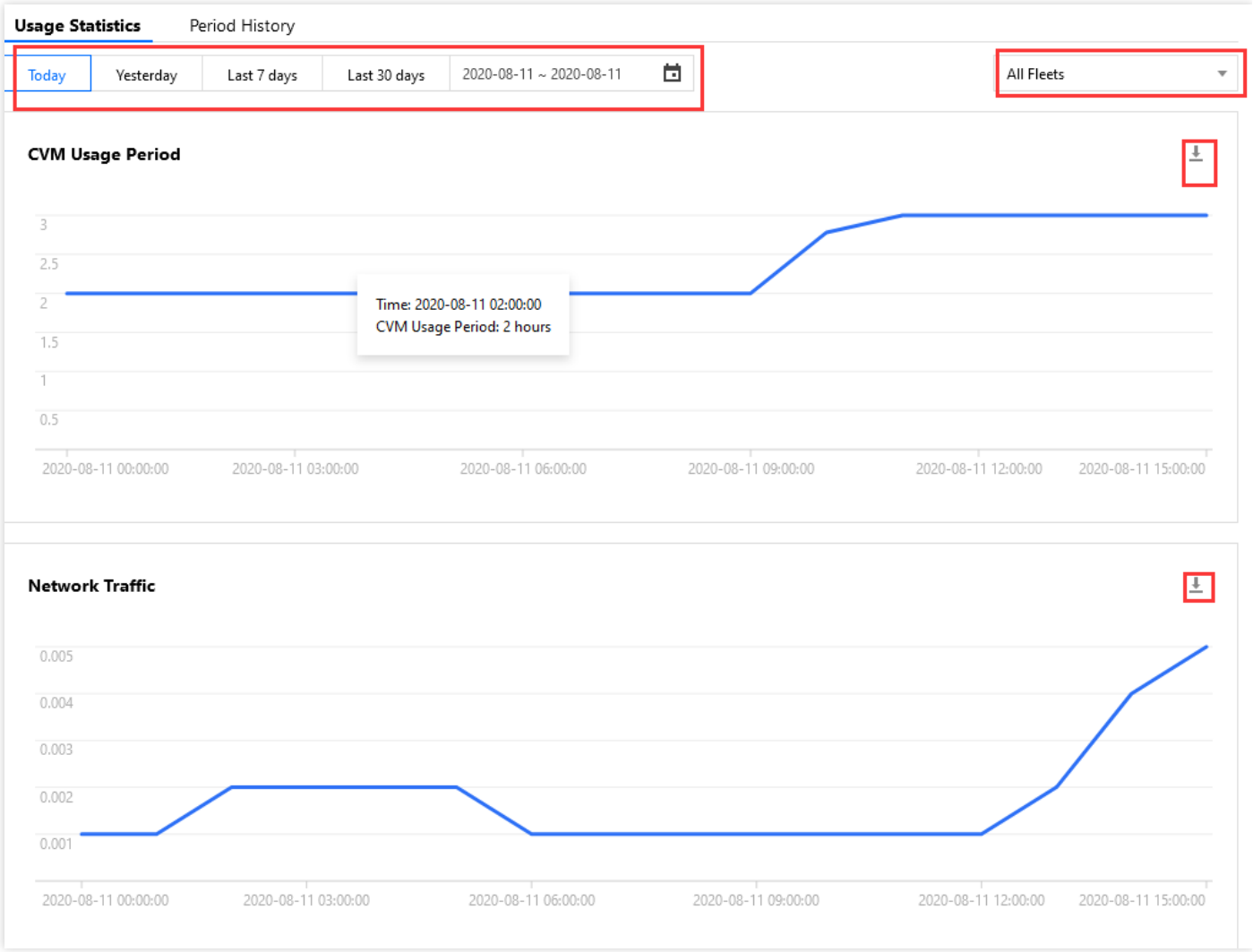
Directions

Statistics

1. Log in to the GSE console and click **Statistics** in the left sidebar.
2. At the top-left corner of the page, select the **Region** where the target server fleet resides to enter the statistics page of the region and view **Usage Statistics** and **History List** information.



- **Usage Statistics**: select **Query time period** and **Fleet name** to get the CVM usage period and network traffic information of the fleet in the specified time period.

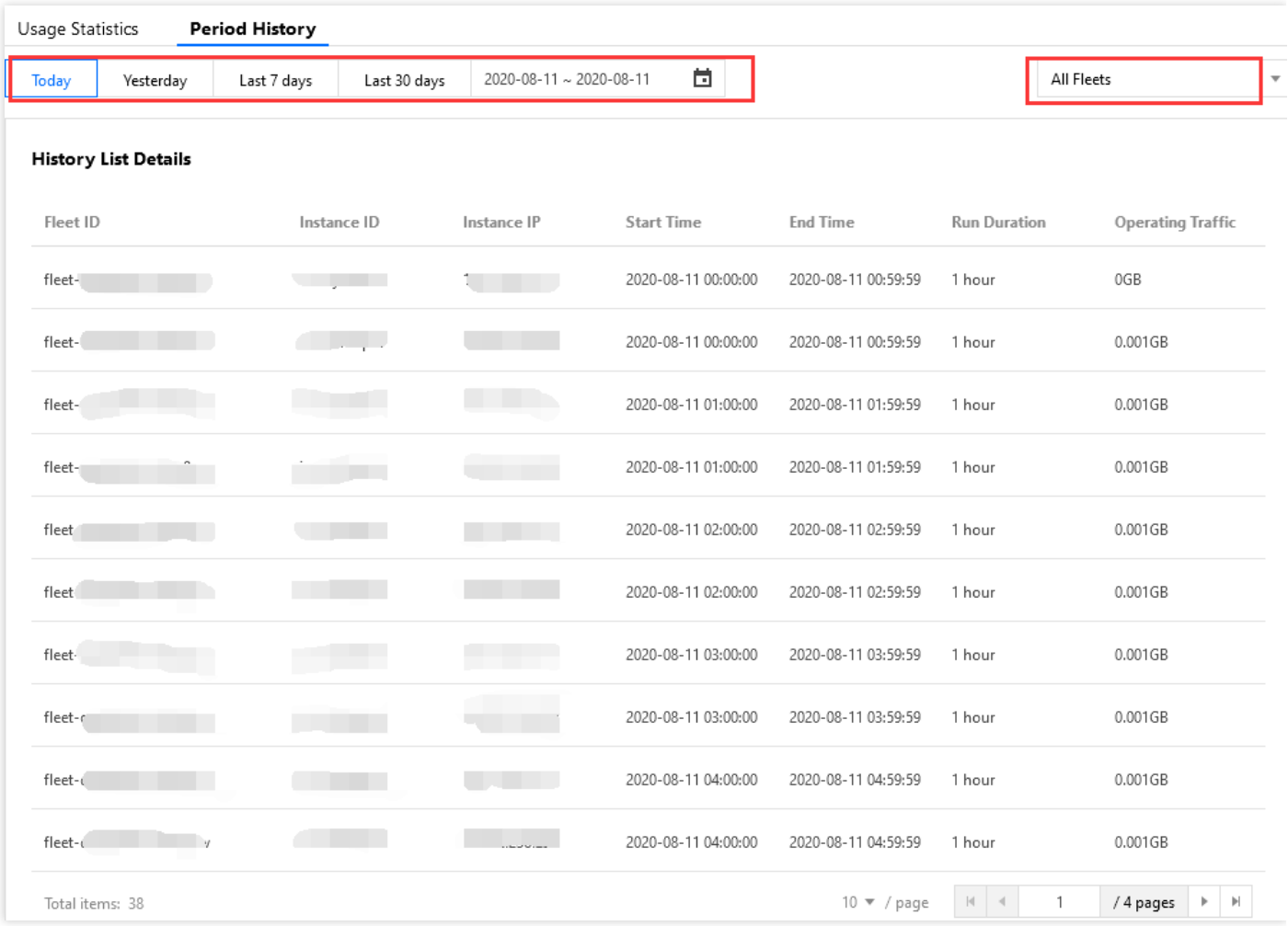


Note :

You can click **Download** to get a detailed data sheet.

- **History List**: select a **Query time period** and **Fleet name** to get the fleet list details in the specified period, including fleet ID, instance ID, instance IP, start time, end time, run duration, and

operating traffic.



Using GPM in GSE

Last updated : 2021-05-18 14:20:49

Overview

You can use GSE with Game Player Matchmaking (GPM) to establish a backend for your battle games. GPM provides player matchmaking services for battle games. With the custom matchmaking rules you set based on simple scripts, it can automatically search players for matchmaking and place game server sessions after the matchmaking is completed.

Directions

When used together with GSE, GPM does not incur extra charges. For more detailed information on **GPM**, see Matchmaking Requiring GSE Resources. You can also independently use GPM services.

Access Management Overview

Last updated : 2021-04-20 14:40:51

Assume that you are using multiple Tencent Cloud services, such as Game Server Elastic-scaling (GSE), VPC and TencentDB. These services are managed by different users who all share your Tencent Cloud account key. Then, the following problems may exist:

- Your key is shared by multiple users, which means your key runs a high risk of being compromised.
- You cannot restrict the access permissions of other users, which poses a security risk due to potential misoperations.

These problems can be eliminated by the use of CAM, which allows you to authorize sub-accounts to manage your different services. By default, a sub-account has no access to GSE service or its resources. To grant a sub-account such access, you need to create a CAM policy. For more information on CAM, see CAM Overview.

A policy is a syntax specification that defines one or more permissions. It allows or denies the access to a specified resource by authorizing a user or a group of users.
For more information on CAM policy elements, please see Element Reference.
For more information on how to use CAM policies, please see Policy.

Note :

If you do not need to manage access permissions to GSE resources for sub-accounts, you can skip this part. This will not affect your understanding and use of other parts of the documentation.

Getting Started

A CAM policy must permit or deny one or more GSE operations. Besides, you must specify some (or all) resources to operate on.

Some GSE APIs support resource-level permissions, which means that you can choose to specify either specific or all resources when calling these APIs.

Task	Link

Basic policy structure	Policy Syntax
Defining operations in a policy	GSE Operations
Defining resources in a policy	GSE Resource Path
Resource-level permissions for GSE	Authorizable Resource Types
Console Demo	Access Control Examples

Authorizable Resource Types

Last updated : 2021-04-20 14:46:45

Game Server Elastic-scaling (GSE) supports resource-level permissions, which means that for certain GSE operations, you can control when users are allowed to perform them, or which specific resources that users are allowed to use. The following will describe the types of resources for which GSE allows permissions.

Note :

Resource-level permissions specify which resources users can operate on.

You can grant permissions for the types of GSE resources below through Cloud Access Management (CAM):

Resource Type	Resource Description Method in Access Policy
GSE Asset Package	`qcs::gse:$region:$account:asset/*`
GSE Alias	`qcs::gse:$region:$account:alias/*`
GSE Fleet	`qcs::gse:$region:$account:fleet/*`
GSE Game Server Session Queue	`qcs::gse:$region:$account:gameServerSessionQueue/*`

GSE Asset Package, GSE Alias, and GSE Fleet list all GSE APIs that support resource-level permissions, and the resources supported by each API. To configure a resource path, you need to replace the variables such as `$region` and `$account` with your own parameters. You can also use the wildcard `*` in a resource path. For examples, see Access Control Examples.

Note :

The GSE APIs not listed below do not support resource-level permissions. For these APIs, you can grant their access permissions as well. However, please note that the `resource` element in their policy statement must be specified as `*`.

GSE Asset Package

API	Resource Path	Description
DeleteAsset	`qcs::gse:$region:$account:asset/*`	Deletes an asset package

	qcs::gse:$region:$account:asset/$AssetId	
DescribeAsset	qcs::gse:$region:$account:asset/* qcs::gse:$region:$account:asset/$AssetId	Retrieves attributes of an asset package
UpdateAsset	qcs::gse:$region:$account:asset/* qcs::gse:$region:$account:asset/$AssetId	Updates attributes of an asset package

GSE Alias

API	Resource Path	Description
DeleteAlias	qcs::gse:$region:$account:alias/* qcs::gse:$region:$account:alias/$AliasId	Deletes an alias
DescribeAlias	qcs::gse:$region:$account:alias/* qcs::gse:$region:$account:alias/$AliasId	Retrieves attributes of an alias
ResolveAlias	qcs::gse:$region:$account:alias/* qcs::gse:$region:$account:alias/$AliasId	Retrieves fleet ID associated with an alias
UpdateAlias	qcs::gse:$region:$account:alias/* qcs::gse:$region:$account:alias/$AliasId	Updates attributes of an alias

GSE Fleet

API	Resource Path	Description
AttachCcnInstances	qcs::gse:$region:$account:asset/* qcs::gse:$region:$account:asset/$AssetId	Associates CCN instances
CreateFleetDemo	qcs::gse:$region:$account:fleet/* qcs::gse:$region:$account:fleet/$FleetId	Creates a fleet demo
DeleteFleet	qcs::gse:$region:$account:fleet/* qcs::gse:$region:$account:fleet/$FleetId	Deletes an empty fleet
DeleteScalingPolicy	qcs::gse:$region:$account:fleet/* qcs::gse:$region:$account:fleet/$FleetId	Deletes a scaling policy
DescribeFleetEvents	qcs::gse:$region:$account:fleet/* qcs::gse:$region:$account:fleet/$FleetId	Retrieves entries from fleet event logs

DescribeFleetPortSettings	qcs::gse:$region:$account:fleet/* qcs::gse:$region:$account:fleet/$FleetId	Retrieves inbound connection permissions for a fleet
DescribeInstances	qcs::gse:$region:$account:fleet/* qcs::gse:$region:$account:fleet/$FleetId	Retrieves fleet instance information
DescribeInstancesExtend	qcs::gse:$region:$account:fleet/* qcs::gse:$region:$account:fleet/$FleetId	Retrieves expansion information on fleet instances
DescribeRuntimeConfiguration	qcs::gse:$region:$account:fleet/* qcs::gse:$region:$account:fleet/$FleetId	Retrieves the runtime configuration of a fleet
DescribeScalingPolicies	qcs::gse:$region:$account:fleet/* qcs::gse:$region:$account:fleet/$FleetId	Retrieves all scaling policies of a fleet
DetachCcnInstances	qcs::gse:$region:$account:fleet/* qcs::gse:$region:$account:fleet/$FleetId	Unassociates CCN instances
GetInstanceAccess	qcs::gse:$region:$account:fleet/* qcs::gse:$region:$account:fleet/$FleetId	Requests remote access permission for instances on a specific fleet
PutScalingPolicy	qcs::gse:$region:$account:fleet/* qcs::gse:$region:$account:fleet/$FleetId	Creates or updates a scaling policy
SetServerWeight	qcs::gse:$region:$account:fleet/* qcs::gse:$region:$account:fleet/$FleetId	Sets server weight
StartFleetActions	qcs::gse:$region:$account:fleet/* qcs::gse:$region:$account:fleet/$FleetId	Starts auto-scaling activities on a fleet
StopFleetActions	qcs::gse:$region:$account:fleet/* qcs::gse:$region:$account:fleet/$FleetId	Suspends auto-scaling activities on a fleet

UpdateDemoResource	qcs::gse:$region:$account:fleet/* qcs::gse:$region:$account:fleet/$FleetId	Modifies the attributes of a service deployment demo
UpdateFleetAttributes	qcs::gse:$region:$account:fleet/* qcs::gse:$region:$account:fleet/$FleetId	Updates general attributes of a fleet
UpdateFleetCapacity	qcs::gse:$region:$account:fleet/* qcs::gse:$region:$account:fleet/$FleetId	Updates fleet capacity
UpdateFleetName	qcs::gse:$region:$account:fleet/* qcs::gse:$region:$account:fleet/$FleetId	Updates a server fleet name
UpdateFleetPortSettings	qcs::gse:$region:$account:fleet/* qcs::gse:$region:$account:fleet/$FleetId	Updates fleet port settings
UpdateFleetVpc	qcs::gse:$region:$account:fleet/* qcs::gse:$region:$account:fleet/$FleetId	Updates the VPC for a game server fleet
UpdateRuntimeConfiguration	qcs::gse:$region:$account:fleet/* qcs::gse:$region:$account:fleet/$FleetId	Updates the regular fleet runtime configuration

GSE Game Server Session Queue

API	Resource Path	Description
DeleteGameServerSessionQueue	qcs::gse:$region:$account:gameServerSessionQueue/* qcs::gse:$region:$account:gameServerSessionQueue/$Name	Deletes a game server session queue
StartGameServerSessionPlacement	qcs::gse:$region:$account:gameServerSessionQueue/* qcs::gse:$region:$account:gameServerSessionQueue/$GameServerSessionQueueName	Starts sending a placement request for a game server session to a game server session queue
UpdateGameServerSessionQueue	qcs::gse:$region:$account:gameServerSessionQueue/*	Updates the attributes of a game

	qcs::gse:$region:$account:gameServerSessionQueue/$Name	server session queue

Access Policy Syntax

Last updated : 2020-08-11 11:17:34

Policy Syntax

A CAM policy is as follows:

```
{
"version":"2.0",
"statement":
[
{
"effect":"effect",
"action":["action"],
"resource":["resource"],
"condition": {"key":{"value"}}
}
]
}
```

- **version**: required. Currently, only "2.0" is allowed.
- **statement**: an element that describes the details on a permission or a permission set defined by other elements including effect, action, resource, and condition. One policy has only one statement.
  - **action**: describes the action to be allowed or denied. An action can be an API (prefixed with `gse:`) or a feature set (a set of specific APIs prefixed with `permid`). This element is required.
  - **resource**: describes the resource to which the permission applies. A resource is described in a six-segment format. Detailed resource definitions vary by product. For more information on how to specify a resource, see the documentation for the product whose resources you are writing a statement for. This element is required.
  - **condition**: describes the condition for the policy to take effect. A condition consists of operator, action key, and action value. A condition value may be time, IP address, etc. Some services allow you to specify additional values in a condition. This element is optional.
  - **effect**: describes whether the result produced by the statement is "allow" or "explicitly deny". This element is required.

GSE Operations

In a CAM policy statement, you can set `action` to any API for any Tencent Cloud service that supports CAM. For GSE APIs, prefix them with `gse:` , such as `gse:AddMatch` or `gse:AddRule` .
To specify multiple actions in a single statement, separate them with a comma as shown below:

```
"action":["gse:action1","gse:action2"]
```

You can also specify multiple actions by using a wildcard, such as all actions that start with "Create" as shown below:

```
"action":["gse:Create*"]
```

To specify all GSE actions, use only the wildcard `*` as follows:

```
"action":["gse:*"]
```

GSE Resource Paths

Each CAM policy statement for GSE is resource-specific with a resource path as shown below:

```
qcs:project_id:service_type:region:account:resource
```

- **project_id**: describes the project information and is only used to enable compatibility with legacy CAM logic. It can be left empty.
- **service_type**: describes the product's abbreviation, such as `gse` .
- *region**: region information, for example, `bj` .
- *account**: the root account of the resource owner, for example, `uin/110702656` .
- **resource**: describes details on the GSE resource, such as `fleet/fleet-28a9refi-ur5pe34j` or `fleet/*` .

For example, you can grant a sub-account the access to the fleet "fleet-28a9refi-ur5pe34j" in Beijing region under your root account by specifying it in the statement, as shown below:

```
"resource":[ "qcs::gse:bj:uin/110702656:fleet/fleet-28a9refi-ur5pe34j"]
```

You can also use the wildcard "*" to specify all resources of a certain type in a specific account as shown below:

```
"resource":[ "qcs::gse::uin/110702656:fleet/*"]
```

If you want to specify all resources or if a specific API does not support resource-level permission control, you can set the value of `resource` to `*` as shown below:

```
"resource": ["*"]
```

To specify multiple resources in one policy, separate them with a comma. In the following example, two resources are specified:

```
"resource":["resource1","resource2"]
```

Access Control Examples

Last updated : 2021-04-20 14:50:55

Overview

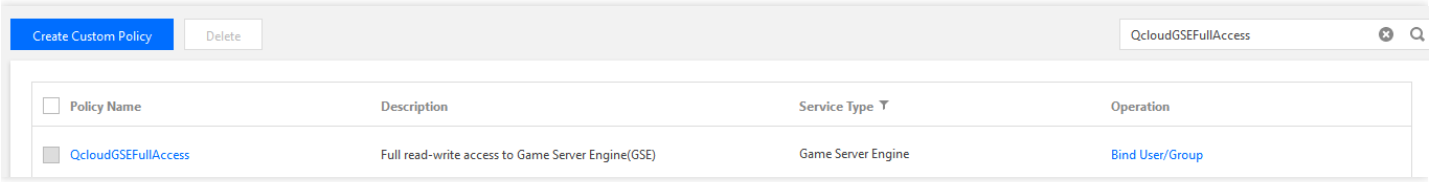
You can grant a user the permission to view and use specific resources in the Game Server Elastic-scaling (GSE) Console by using CAM policies. The examples below show how to do so.

Directions

Full access policy for GSE

To grant a user full permission to create and manage all GSE resources, associate the `QcloudGSEFullAccess` policy with the user.
The detailed steps are as follows:

1. Go to the Policies page in the console, and click **Service Type**.
2. Search for this policy by selecting **Game Server Elastic-scaling** from the drop-down list, or by directly using the search box in the top right corner.



The policy syntax is as follows:

```
{
"version": "2.0",
"statement": [
{
"action": [
"gse:*"
],
"resource": "*",
"effect": "allow"
}
```

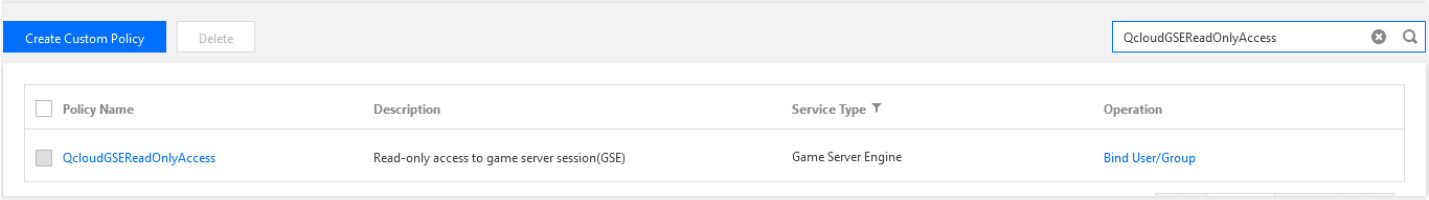
```
]
}
```

The above policy allows the user full access permission to GSE, including activating service, viewing statistics, and accessing asset packages, aliases, fleets, game server session queues, and any other resources.

Read-only policy for GSE

To grant a user permission to query any GSE resources, but not create, delete, or modify them, associate the `QcloudCDBInnerReadOnlyAccess` policy with the user.
The detailed steps are as follows:

1. Go to the Policies page in the console, and click **Service Type**.
2. Search for this policy by selecting **Game Server Elastic-scaling** from the drop-down list, or by directly using the search box in the top right corner.



The policy syntax is as follows:

```
{
"version": "2.0",
"statement": [
{
"action": [
"gse:Get*",
"gse:Describe*",
"gse:List*",
"gse:Resolve*",
"gse:Search*",
"gse:Check*"
],
"resource": "*",
"effect": "allow"
}
]
}
```

Policy for querying GSE service status

To grant a user the permission to query the status of GSE service, you can include the following operation in your policy, and then associate the policy with the user.

Name	Description
CheckOpenStatus	Checks whether the service is activated or not

The detailed steps are as follows:

1. Create a custom policy for querying GSE service status as instructed in Policy. The example policy syntax is as follows:

```
{
"version": "2.0",
"statement": [
{
"action": [
"gse:CheckOpenStatus"
],
"resource": "*",
"effect": "allow"
}
]
}
```

2. Find the created policy and click **Bind User/Group** in the "Operation" column.
3. In the pop-up window, select the user/group you want to authorize and click **OK**.

⚠ Note :

To initiate a GSE API request always requires the permission to the `CheckOpenStatus` API. Therefore, you must have `CheckOpenStatus` included in your custom policy. To do so, you can manually add it to each of your custom policies, or create a separate custom policy for this API like the example above, and associate it with any specific user.

Policy for activating GSE service

To grant a user the permission to activate GSE service, you can include the following operation in your policy, and then associate the policy with the user.

Name	Description

SetOpenStatus	Activates GSE service

The detailed steps are as follows:

1. Create a custom policy for activating GSE service as instructed in Policy. The example policy syntax is as follows:

```
{
"version": "2.0",
"statement": [
{
"action": [
"gse:SetOpenStatus"
],
"resource": "*",
"effect": "allow"
}
]
}
```

2. Find the created policy and click **Bind User/Group** in the "Operation" column.
3. In the pop-up window, select the user/group you want to authorize and click **OK**.

Policy for viewing GSE statistics

To grant a user the permission to view GSE statistics, you can include the following operations in your policy, and then associate the policy with the user.

Name	Description
DescribeFleetStatisticDetails	Queries details on fleet statistics
DescribeFleetStatisticFlows	Queries fleet usage statistics
DescribeFleetStatisticSummary	Queries a summary of fleet statistics
ListFleets	Retrieves a fleet list

The detailed steps are as follows:

1. Create a custom policy for viewing GSE statistics as instructed in Policy. The example policy syntax is as follows:

```
{
"version": "2.0",
"statement": [
{
"action": [
"gse:ListFleets",
"gse:DescribeFleetStatisticDetails",
"gse:DescribeFleetStatisticFlows",
"gse:DescribeFleetStatisticSummary"
],
"resource": "*",
"effect": "allow"
}
]
}
```

2. Find the created policy and click **Bind User/Group** in the "Operation" column.
3. In the pop-up window, select the user/group you want to authorize and click **OK**.

Policy for GSE matching

To grant a user the permission for matching on GSE, you can include the following operation in your policy, and then associate it with the user.

Name	Description
StartMatchPlacement	Starts to match and place a game server session

The detailed steps are as follows:

1. Create a custom policy for matching on GSE as instructed in Policy. The example policy syntax is as follows:

```
{
"version": "2.0",
"statement": [
{
"action": "gse:StartMatchPlacement",
"resource": "*",
"effect": "allow"
}
]
}
```

2. Find the created policy and click **Bind User/Group** in the "Operation" column.
3. In the pop-up window, select the user/group you want to authorize and click **OK**.

Policy for listing GSE resources

To grant a user the permission to retrieve GSE resources lists, you can include the following operations in your policy, and then associate the policy with the user.

Name	Description
DescribeAssets	Retrieves a list of asset packages
DescribeGameServerSessionQueues	Queries game server session queues
DescribeFleetAttributes	Retrieves fleet attributes (including status)
DescribeFleetCapacity	Retrieves the current fleet capacity settings
DescribeFleetUtilization	Retrieves fleet utilization statistics
DescribeInstanceTypes	Retrieves a list of server instance types
DescribeUserQuotas	Retrieves quotas for modules
ListAliases	Retrieves an alias list
ListFleets	Retrieves a fleet list

The detailed steps are as follows:

1. Create a custom policy for listing GSE resources as instructed in Policy. The example policy syntax is as follows:

```
{
"version": "2.0",
"statement": [
{
"action": [
"gse:DescribeFleetUtilization",
"gse:DescribeInstanceTypes",
"gse:DescribeFleetAttributes",
"gse:DescribeFleetCapacity",
"gse:DescribeUserQuotas",
"gse:DescribeAssets",
"gse:DescribeGameServerSessionQueues"
"gse:ListAliases",
```

```
"gse:ListFleets"
],
"resource": "*",
"effect": "allow"
}
]
}
```

2. Find the created policy and click **Bind User/Group** in the "Operation" column.
3. In the pop-up window, select the user/group you want to authorize and click **OK**.

Full access policy for GSE excluding setting service status and viewing statistics

To grant a user full access permission to GSE, excluding setting service status and viewing statistics, you can create a custom policy to associate with the user using a policy syntax as follows:

```
{
"version": "2.0",
"statement": [
{
"action": [
"gse:*"
],
"resource": "*",
"effect": "allow"
},
{
"action": [
"gse:SetOpenStatus",
"gse:DescribeFleetStatisticDetails",
"gse:DescribeFleetStatisticFlows",
"gse:DescribeFleetStatisticSummary"
],
"resource": "*",
"effect": "deny"
}
]
}
```

Policy for accessing GSE asset packages

To grant a user the permission to access all asset packages on GSE in a specific account, e.g. 110702656, you can include the following operations in your policy, and then associate it with the user.

Name	Description
DeleteAsset	Deletes an asset package
DescribeAsset	Retrieves asset package attributes
UpdateAsset	Updates asset package attributes

The detailed steps are as follows:

1. Create a custom policy for accessing all asset packages on GSE in account 110702656 as instructed in Policy. The example policy syntax is as follows:

```
{
"version": "2.0",
"statement": [
{
"action": [
"gse:DeleteAsset",
"gse:DescribeAsset",
"gse:UpdateAsset"
],
"resource": "qcs::gse::uin/110702656:asset/*",
"effect": "allow"
}
]
}
```

2. Find the created policy and click **Bind User/Group** in the "Operation" column.
3. In the pop-up window, select the user/group you want to authorize and click **OK**.

To allow the user access to a specific asset package, e.g. AssetId=asset-eiu39dki-3o0dk2dk, in account 110702656, configure the value of `resource` to:

```
qcs::gse::uin/110702656:asset/asset-eiu39dki-3o0dk2dk
```

Policy for accessing GSE aliases

To grant a user the permission to access all aliases on GSE in a specific account, e.g. 110702656, you can include the following operations in your policy, and then associate it with the user.

Name	Description
DeleteAlias	Deletes an alias

DescribeAlias	Retrieves alias attributes
ResolveAlias	Retrieves the fleet ID associated with an alias
UpdateAlias	Updates alias attributes

The detailed steps are as follows:

1. Create a custom policy for accessing all aliases on GSE in account 110702656 as instructed in Policy. The example policy syntax is as follows:

```
{
"version": "2.0",
"statement": [
{
"action": [
"gse:DeleteAlias",
"gse:DescribeAlias",
"gse:ResolveAlias",
"gse:UpdateAlias"
],
"resource": "qcs::gse::uin/110702656:alias/*",
"effect": "allow"
}
]
}
```

2. Find the created policy and click **Bind User/Group** in the "Operation" column.
3. In the pop-up window, select the user/group you want to authorize and click **OK**.

To allow the user access to a specific alias, e.g. AliasId=alias-wk30dke9-ovr93dk3, in account 110702656, configure the value of `resource` to:

```
qcs::gse::uin/110702656:asset/alias-wk30dke9-ovr93dk3
```

Policy for accessing GSE fleets

To grant a user the permission to access all fleets on GSE in a specific account, e.g. 110702656, you can include the following operations in your policy, and then associate it with the user.

Name	Description
AttachCcnInstances	Associates CCN instances

CreateFleetDemo	Creates a fleet demo
DeleteFleet	Deletes an empty fleet
DeleteScalingPolicy	Deletes a scaling policy
DescribeFleetEvents	Retrieves entries from fleet event logs
DescribeFleetPortSettings	Retrieves the inbound connection permissions for a fleet
DescribeInstances	Retrieves fleet instance information
DescribeInstancesExtend	Retrieves the scaling information of fleet instances
DescribeRuntimeConfiguration	Retrieves a fleet runtime configuration
DescribeScalingPolicies	Retrieves all fleet scaling policies
DetachCcnInstances	Unassociates CCN instances
GetInstanceAccess	Requests remote access to the specified fleet instance
PutScalingPolicy	Creates or updates a scaling policy
SetServerWeight	Sets server weight
StartFleetActions	Starts auto-scaling activities on a fleet
StopFleetActions	Suspends auto-scaling activities on a fleet
UpdateDemoResource	Modifies the attributes of a service deployment demo
UpdateFleetAttributes	Updates general attributes of a fleet
UpdateFleetCapacity	Updates fleet capacity
UpdateFleetName	Updates a server fleet name
UpdateFleetPortSettings	Updates fleet port settings
UpdateFleetVpc	Updates the VPC for a game server fleet
UpdateRuntimeConfiguration	Updates the regular fleet runtime configuration

The detailed steps are as follows:

1. Create a custom policy for accessing all fleets on GSE in account 110702656 as instructed in Policy. The example policy syntax is as follows:

```
{
"version": "2.0",
"statement": [
{
"action": [
"gse:AttachCcnInstances",
"gse:CreateFleetDemo",
"gse:DeleteFleet",
"gse:DeleteScalingPolicy",
"gse:DescribeFleetEvents",
"gse:DescribeFleetPortSettings",
"gse:DescribeInstances",
"gse:DescribeInstancesExtend",
"gse:DescribeRuntimeConfiguration",
"gse:DescribeScalingPolicies",
"gse:DetachCcnInstances",
"gse:GetInstanceAccess",
"gse:PutScalingPolicy",
"gse:SetServerWeight",
"gse:StartFleetActions",
"gse:UpdateDemoResource",
"gse:UpdateFleetAttributes",
"gse:UpdateFleetCapacity",
"gse:UpdateFleetName",
"gse:UpdateFleetPortSettings",
"gse:UpdateFleetVpc",
"gse:UpdateRuntimeConfiguration"
],
"resource": "qcs::gse::uin/110702656:fleet/*",
"effect": "allow"
}
]
}
```

2. Find the created policy and click **Bind User/Group** in the "Operation" column.
3. In the pop-up window, select the user/group you want to authorize and click **OK**.

To allow the user access to a specific fleet, e.g. FleetId=fleet-28a9refi-ur5pe34j, in account 110702656, configure the value of `resource` to:

```
qcs::gse::uin/110702656:fleet/fleet-28a9refi-ur5pe34j
```

Policy for accessing GSE game server session queues

To grant a user the permission to access all game server session queues on GSE in a specific account, e.g. 110702656, you can include the following operations in your policy, and then associate it with the user.

Name	Description
DeleteGameServerSessionQueue	Deletes a game server session queue
StartGameServerSessionPlacement	Starts sending a game server session placement request to the game server session queue
UpdateGameServerSessionQueue	Updates the attributes of a game server session queue

The detailed steps are as follows:

1. Create a custom policy for accessing all game server session queues on GSE in account 110702656 as instructed in Policy. The example policy syntax is as follows:

```
{
"version": "2.0",
"statement": [
{
"action": [
"gse:DeleteGameServerSessionQueue",
"gse:StartGameServerSessionPlacement",
"gse:UpdateGameServerSessionQueue"
],
"resource": "qcs::gse::uin/110702656:gameServerSessionQueue/*",
"effect": "allow"
}
]
}
```

2. Find the created policy and click **Bind User/Group** in the "Operation" column.
3. In the pop-up window, select the user/group you want to authorize and click **OK**.

To allow the user access to a specific game server session queue in account 110702656, e.g. "GameServerSessionQueueName=queue-ciu3c0de-c09dk3do" in the `StartGameServerSessionPlacement` input parameters, or "Name=queue-ciu3c0de-c09dk3do" in the `DeleteGameServerSessionQueue/UpdateGameServerSessionQueue` input parameters, configure the value of `resource` to:

```
qcs::gse::uin/110702656:gameServerSessionQueue/queue-ciu3c0de-c09dk3do
```

Policy for accessing Demo

To grant a user the permission to access the Demo for GSE, you can include the following operations in your policy, and then associate it with the user.

Name	Description
CreateAssetAuto	Creates an asset package automatically
CreateFleetDemo	Creates a fleet demo
CreateGameServerSession	Creates a game server session
DescribeFleetAttributes	Retrieves fleet attributes (including status)
DescribeGameServerSessions	Queries a list of game server sessions
DescribeDemoResource	Retrieves the attributes of a service deployment demo
JoinGameServerSession	Joins a game server session
UpdateDemoResource	Modifies the attributes of a service deployment demo

The detailed steps are as follows:

1. Create a custom policy for accessing the Demo for GSE as instructed in Policy. The example policy syntax is as follows:

```
{
"version": "2.0",
"statement": [
{
"action": [
"gse:CreateAssetAuto",
"gse:UpdateDemoResource",
"gse:CreateFleetDemo",
"gse:DescribeDemoResource",
"gse:DescribeFleetAttributes",
"gse:DescribeGameServerSessions",
"gse:CreateGameServerSession",
"gse:JoinGameServerSession"
],
"resource": "*",
"effect": "allow"
}
```

```
]
}
```

2. Find the created policy and click **Bind User/Group** in the "Operation" column.
3. In the pop-up window, select the user/group you want to authorize and click **OK**.