solution

Deploying a Scalable HPC Cluster with Slurm

Issue 1.0.0

Date 2024-04-30

HUAWEI TECHNOLOGIES CO., LTD.

Trademarks and Permissions


and other Huawei trademarks are trademarks of Huawei Technologies Co., Ltd.
All other trademarks and trade names mentioned in this document are the property of their respective holders.

Notice

The purchased products, services and features are stipulated by the contract made between Huawei and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

Security Declaration

Vulnerability

Huawei's regulations on product vulnerability management are subject to the *Vul. Response Process.* For details about this process, visit the following web page:
https://www.huawei.com/en/psirt/vul-response-process
For vulnerability information, enterprise customers can visit the following web page:
https://securitybulletin.huawei.com/enterprise/en/security-advisory

Contents

1 Solution Overview

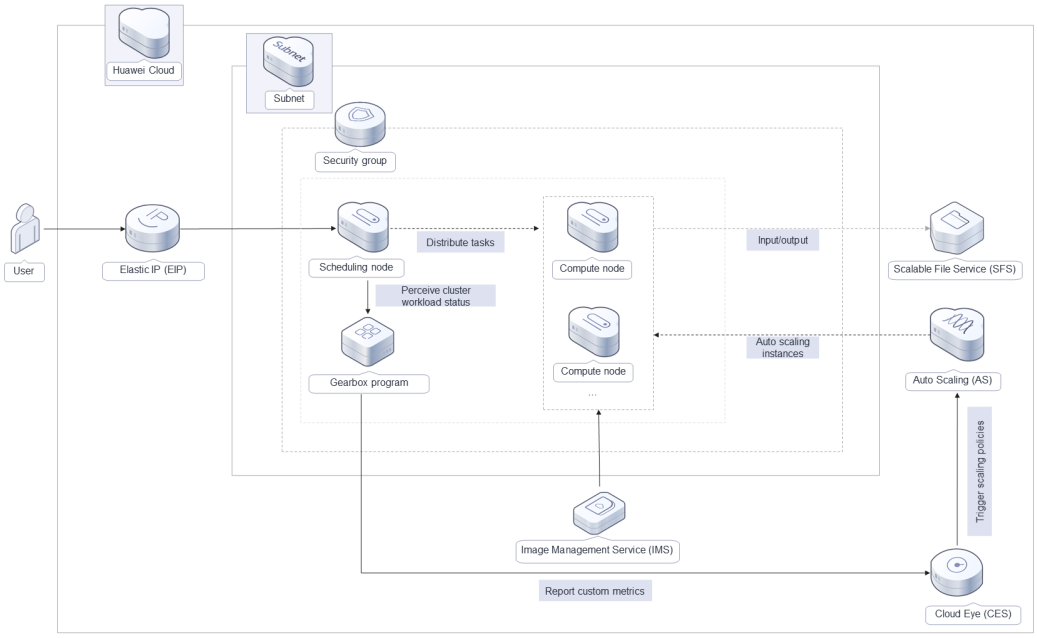
Scenarios

This solution helps you quickly set up a scalable HPC environment on Huawei Cloud based on the open-source software Slurm and Huawei's open-source Gearbox. Slurm is configured to run in "configless" mode for cloud servers functioning as compute nodes. The Gearbox program interconnects with Huawei Cloud Auto Scaling and Cloud Eye to monitor the job status of a Slurm cluster and automatically scale in or out cloud servers in the Slurm cluster in real time. In addition, new cloud servers are automatically registered with and added to the cluster, or cloud servers are automatically deregistered from the cluster and then destroyed.

Solution Architecture

The solution architecture is illustrated below.

Figure 1-1 Architecture



This solution will:

- Create two Linux Elastic Cloud Servers (ECSs), install the open-source software Slurm, install the Gearbox program on the scheduling node, and configure the Java environment.
- Create one EIP for internal and external communication.
- Create security groups and configure rules to control access to ECSs so as to secure the ECS environment.
- Use Image Management Service (IMS) to prepare the initialization environment for compute node servers during auto scaling.
- Use Auto Scaling to create and configure an auto scaling group as well as define scaling policies to automatically scale in or out cluster resources.
- Use Cloud Eye for resource monitoring. The Gearbox program monitors the job status, calculates the workload value of custom metrics, and reports the metrics to Cloud Eye.
- Use Scalable File Service (SFS) to mount SFS file systems to the ECSs to provide shared file storage for clusters.

Advantages

- Auto scaling

  In this solution, auto scaling groups are configured and the Gearbox program is built in the server that functions as the scheduling node. The program periodically monitors cluster metrics, summarizes metric data, and reports the data to Cloud Eye. Cloud Eye alarm rules then trigger auto scaling, reducing costs.

- Personalized customization

  This solution and the built-in Gearbox program are both open-source and free for commercial use. You can also make custom development based on source code.

- Easy deployment

  In just a few clicks, you can easily deploy a scalable HPC cluster.

Constraints

- Before deploying this solution, register a HUAWEI ID, enable Huawei Cloud services, and complete real-name authentication. If you select the yearly/monthly billing mode, ensure that your account has sufficient balance. If you do not have sufficient balance, you can go to the **Billing Center** to manually pay for the order.
- Before deploying this solution, ensure that your account has sufficient IAM permissions. For details, see **(Optional) Creating the rf_admin_trust Agency**.
- Ensure that you have sufficient quotas. Specifically, log in to the Huawei Cloud management console and choose **Resources** > **My Quotas** to check your quotas. If the quotas are insufficient, submit a service ticket to increase the quotas.
  - Compute: number of ECSs, number of CPU cores, and RAM capacity
  - Storage: Elastic Volume Service (EVS) and Scalable File Service (SFS)

- Network: VPC, subnets, EIPs, and security groups

2 Resource Planning and Costs

This solution will deploy the resources listed in the following table. The costs are only estimates and may differ from the final prices. For details, see **Huawei Cloud Pricing**.

Table 2-1 Resource planning and costs (pay-per-use)

Huawei Cloud Service	Example Configuration	Estimated Monthly Cost
Elastic Cloud Server (ECS) (scheduling node)	• Pay-per-use: $0.17 USD/hour • Region: AP-Singapore • Billing Mode: Pay-per-use • Type: X86 | General computing-plus | c6s.xlarge.2 | 4vCPUs | 8 GiB • Image: CentOS 7.9 64bit • System Disk: High I/O | 100 GB • Quantity: 1	$ 118.80 USD
Elastic Cloud Server (ECS) (compute node)	• Pay-per-use: $1.27 USD/hour • Region: AP-Singapore • Billing Mode: Pay-per-use • Type: X86 | General computing-plus | c6s.8xlarge.2 | 32vCPUs | 64GiB • Image: CentOS 7.9 64bit • System Disk: High I/O | 100 GB • Quantity: 1	$ 915.12 USD

Huawei Cloud Service	Example Configuration	Estimated Monthly Cost
Scalable File Service (SFS)	• Pay-per-use: $0.06 USD/hour • Region: AP-Singapore • Billing Mode: Pay-per-use • SFS turbo standard | 500GB	$ 46.08 USD
Elastic IP (EIP)	• Pay-per-use: $0.13 USD/hour • Region: AP-Singapore • Billing Mode: Pay-per-use • Routing Type: Dynamic BGP • Billed By: Bandwidth • Bandwidth: 5 Mbit/s • EIP Quantity: 1	$ 90.00 USD
Total		**$1170 USD + Price of public network traffic**

Table 2-2 Resource planning and costs (yearly/monthly)

Huawei Cloud Service	Example Configuration	Estimated Monthly Cost
Elastic Cloud Server (ECS) (scheduling node)	• Region: AP-Singapore • Billing Mode: Yearly/Monthly • Type: X86 | General computing-plus | c6s.xlarge.2 | 4vCPUs | 8 GiB • Image: CentOS 7.9 64bit • System Disk: High I/O | 100 GB • Quantity: 1	$ 91.82 USD

Huawei Cloud Service	Example Configuration	Estimated Monthly Cost
Elastic Cloud Server (ECS) (compute node)	• Region: AP-Singapore • Billing Mode: Yearly/ Monthly • Type: X86 | General computing-plus | c6s.8xlarge.2 | 32vCPUs | 64GiB • Image: CentOS 7.9 64bit • System Disk: High I/O | 100 GB • Quantity: 1	$ 701.52 USD
Scalable File Service (SFS)	• Region: AP-Singapore • Billing Mode: Yearly or Monthly (The default billing mode is pay-per-use. You need to manually **change the billing mode to yearly/monthly**. • SFS turbo standard | 500GB	$ 40.00 USD
Elastic IP (EIP)	• Region: AP-Singapore • Billing Mode: Yearly/ Monthly • Routing Type: Dynamic BGP • Billed By: Bandwidth • Bandwidth: 5 Mbit/s • Quantity: 1	$ 57.00 USD
Total	-	**$890.34 USD + Price of public network traffic**

3 Procedure

3.1 Preparations

(Optional) Creating the rf_admin_trust Agency

Step 1 Access the Huawei Cloud official website, log in to the **console**, hover your mouse over the account name in the upper right corner, and choose **Identity and Access Management**.

Figure 3-1 Huawei Cloud console

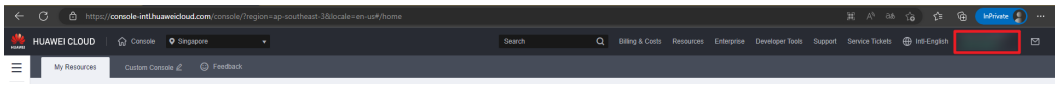
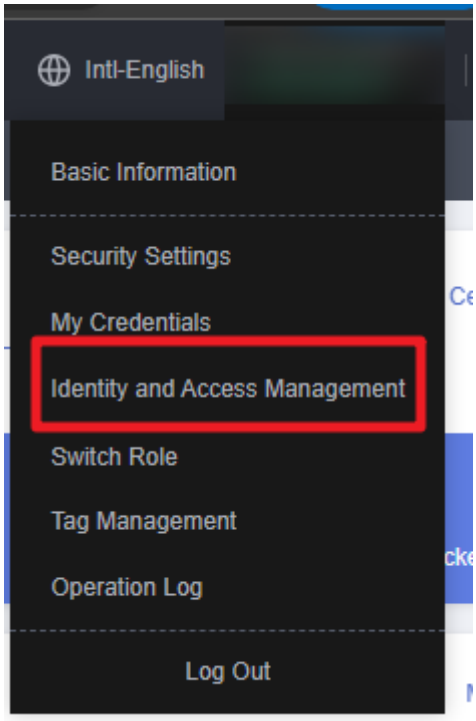
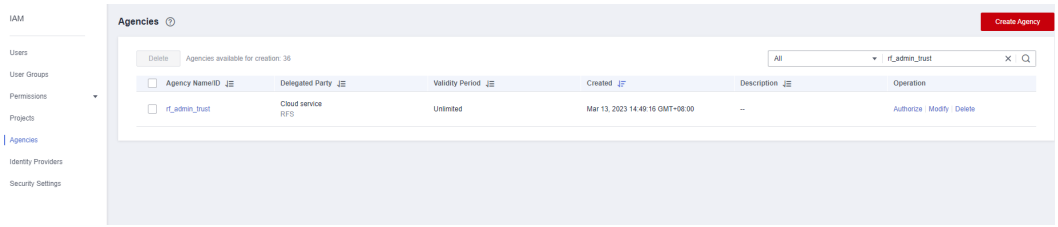


Figure 3-2 Identity and Access Management



Step 2 Choose **Agencies** in the left navigation pane and search for the **rf_admin_trust** agency.

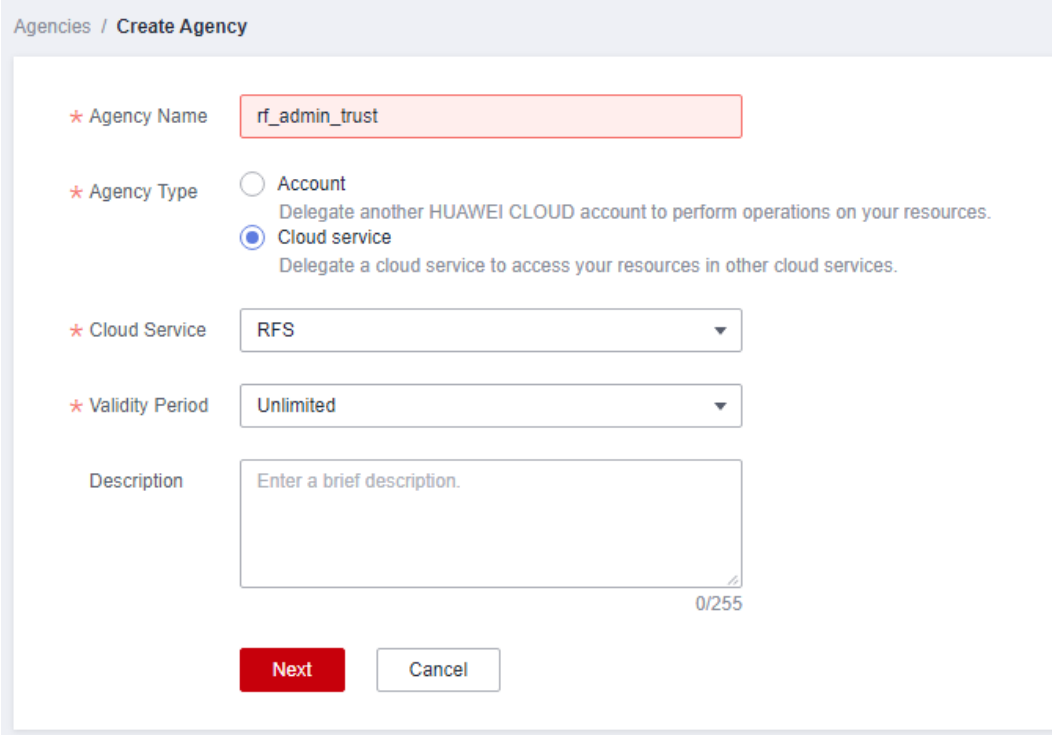
Figure 3-3 Agency list



- If the agency is found, skip the following steps.
- If the agency is not found, perform the following steps.

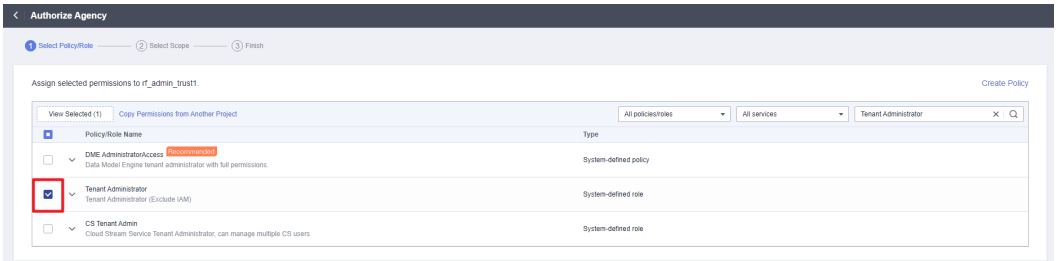
Step 3 Click **Create Agency** in the upper right corner of the page. On the displayed page, set **Agency Name** to **rf_admin_trust**, **Agency Type** to **Cloud service**, **Cloud Service** to **RFS**, and click **Next**.

Figure 3-4 Creating an agency



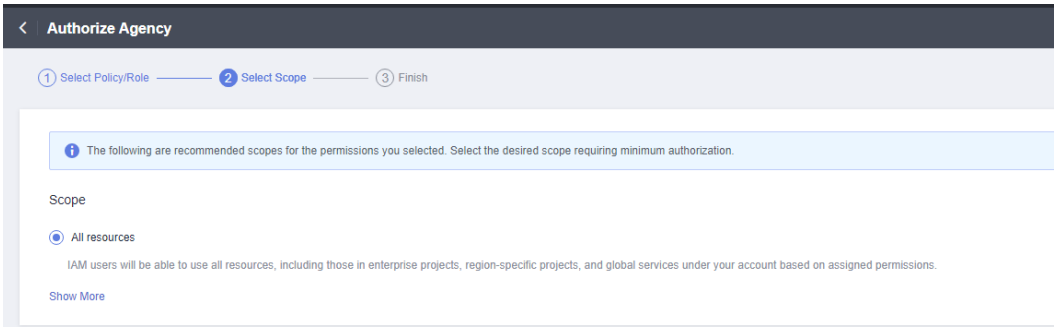
Step 4 Search for **Tenant Administrator**, select it in the search results, and click **Next**.

Figure 3-5 Selecting a policy



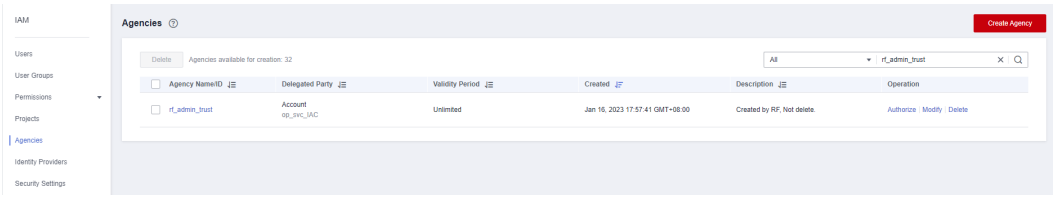
Step 5 Select **All resources** and click **OK**.

Figure 3-6 Setting the authorization scope



Step 6 Check that the **rf_admin_trust** agency is displayed in the agency list.

Figure 3-7 Agency list



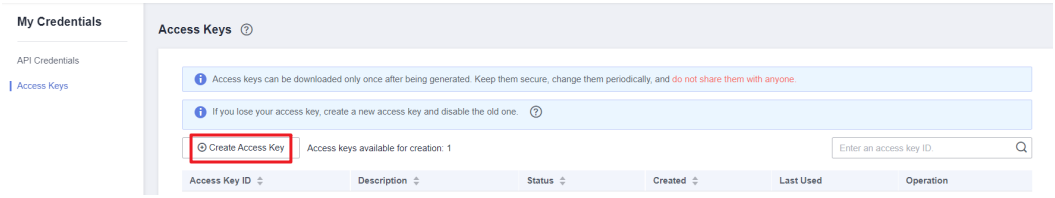
----End

Creating an AK/SK Pair

Before deploying this solution, you need to configure an AK/SK pair on the Huawei Cloud console and create an account key pair for logging in to ECSs.

Step 1 Follow the instructions in the **official documentation** to create an AK on the **Access Keys** page and download the SK. The AK/SK pair will be used as configuration items for starting Gearbox. (After this solution is successfully deployed, you need to delete the AK/SK pair and create a new pair.)

Figure 3-8 Creating an AK/SK pair



Step 2 Follow the instructions in the **official documentation** to create an account key pair on the **Kay Pair Service** page. The account key pair will be used as the login key of the added ECS in subsequent **quick deployment**.

Figure 3-9 Creating a key pair

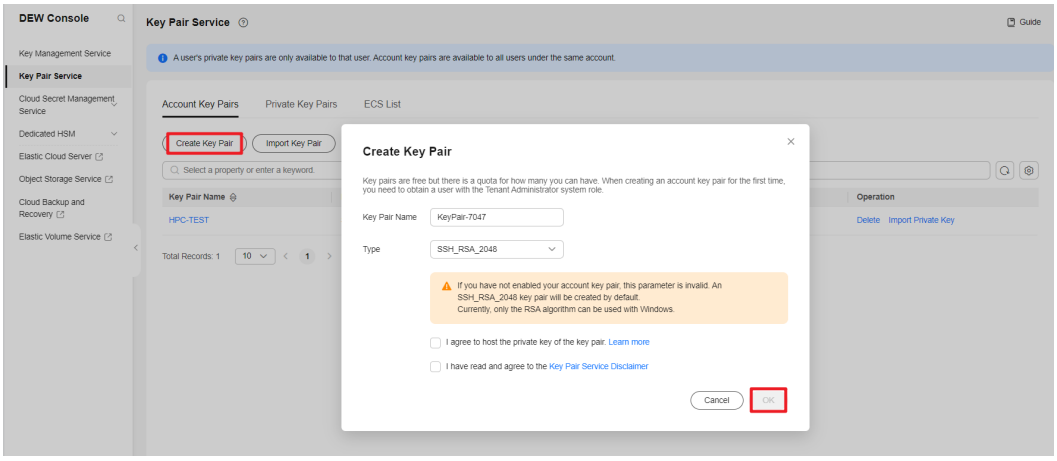
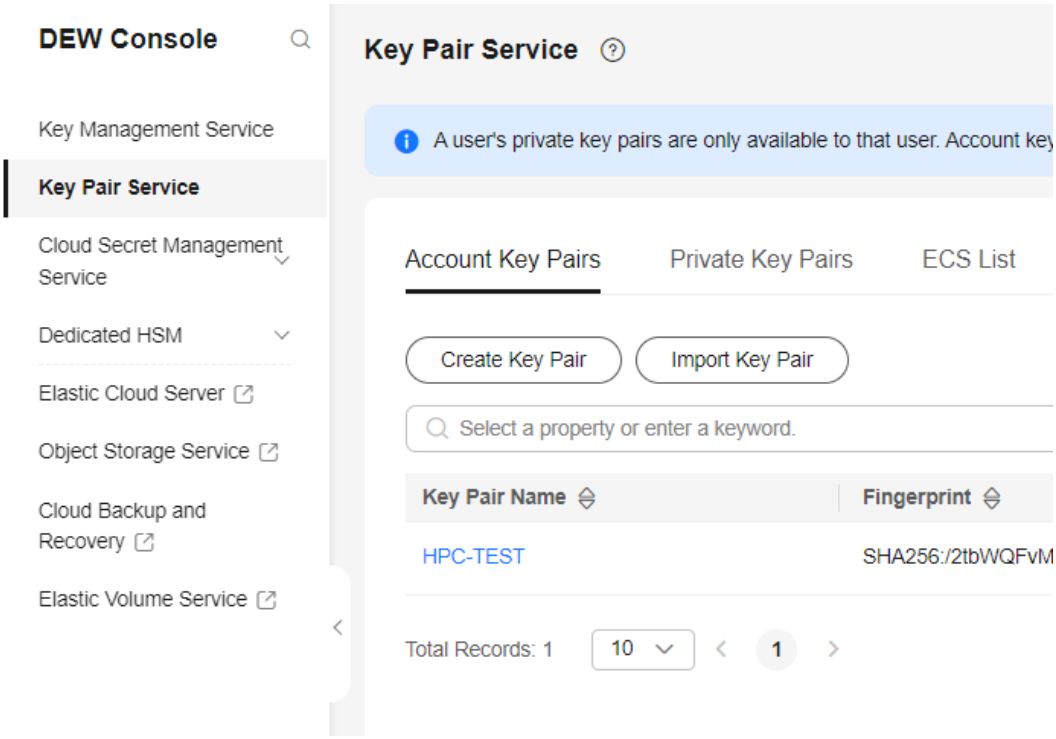


Figure 3-10 Key pair created



----End

3.2 Quick Deployment

This section describes how to quickly deploy this solution.

Table 3-1 Parameter

Name	Type	Mandatory	Description	Default Value
vpc_name	string	Yes	Virtual Private Cloud (VPC) name. This template uses a newly created VPC. The VPC name must be unique. It can contain 1 to 54 characters. Only letters, digits, underscores (_), hyphens (-), and periods (.) are allowed.	scalable-hpc-cluster-with-slurm-demo
security_group_name	string	Yes	Security group name. This template uses a newly created security group. It can contain 1 to 64 characters. Only letters, digits, underscores (_), hyphens (-), and periods (.) are allowed.	scalable-hpc-cluster-with-slurm-demo
bandwidth_size	number	Yes	Bandwidth size, in Mbit/s. Value range: 1-2,000	5

Name	Type	Mandatory	Description	Default Value
ecs_name	string	Yes	ECS name. It must be unique. The name is in the **{*ecs_name*}-master** or **{*ecs_name*}-node** format. It can contain 1 to 57 characters. Only lowercase letters, digits, and hyphens (-) are allowed.	scalable-hpc-cluster-with-slurm-demo
ecs_password	string	Yes	Initial password of an ECS. It can contain 8 to 26 characters, and it must include at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters ($!@%-_=+[]:./^,{}?). The password cannot contain the username or the username spelled backwards. The default administrator account is **root**.	Left blank

Name	Type	Mandatory	Description	Default Value
ecs_master_flavor	string	Yes	The flavor of an ECS that can function as a scheduling node. For more flavors, see **A Summary List of x86 ECS Specifications**.	c6s.xlarge.2
ecs_node_flavor	string	Yes	The flavor of an ECS that can function as a compute node. For more flavors, see **A Summary List of x86 ECS Specifications**.	c6s.8xlarge.2
keypair_name	string	Yes	The name of the existing SSH key pair. Only account key pairs under **Key Pair Service** are supported. For details, see **Key Pair Management**.	Left blank

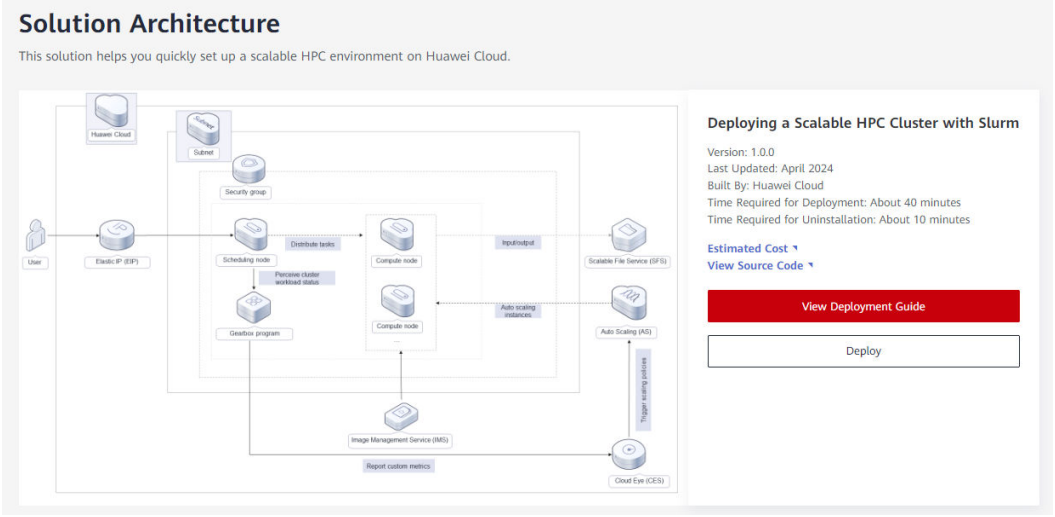
Name	Type	Mandatory	Description	Default Value
sfs_turbo_name	string	Yes	The name of an SFS file. It can contain 4 to 64 characters, and it must start with a letter, case-insensitive. Only letters, digits, hyphens (-), and underscores (_) are allowed.	scalable-hpc-cluster-with-slurm-demo
sfs_turbo_size	number	Yes	The size of an SFS file, in GB. SFS file systems provide shared file storage for clusters. Value range: 500-32,768	500
as_name	string	Yes	The name of an auto scaling resource. This template uses a newly created auto scaling group to provide auto scaling for clusters. The name must be unique. It can contain 1 to 50 characters. Only letters, digits, and hyphens (-) are allowed.	scalable-hpc-cluster-with-slurm-demo

Name	Type	Mandatory	Description	Default Value
charge_mode	string	Yes	Billing mode. It can be **prePaid** (yearly/monthly) or **postPaid** (pay-per-use).	postPaid
charge_period_unit	string	Yes	Unit of a subscription term. This parameter is mandatory only when **charging_mode** is set to **prePaid**. Value range: **month** or **year**	month
charge_period	number	Yes	Subscription term. This parameter is mandatory only when **charging_mode** is set to **prePaid**. Value range: • 1-9 (**charging_period_unit** set to **month**) • 1-3 (**charging_period_unit** set to **year**)	1

Name	Type	Mandatory	Description	Default Value
AK	string	Yes	The Access Key ID (AK) created for configuring Gearbox. For details about how to create and obtain an access key, see **Step 1**.	Left blank
SK	string	Yes	The Secret Access Key (SK) created for configuring Gearbox. For details about how to create and obtain an access key, see **Step 1**.	Left blank
project_id	string	Yes	The project ID for the region where the solution is deployed. For details about how to obtain the project ID, see **Identity and Access Management > Projects**.	Left blank

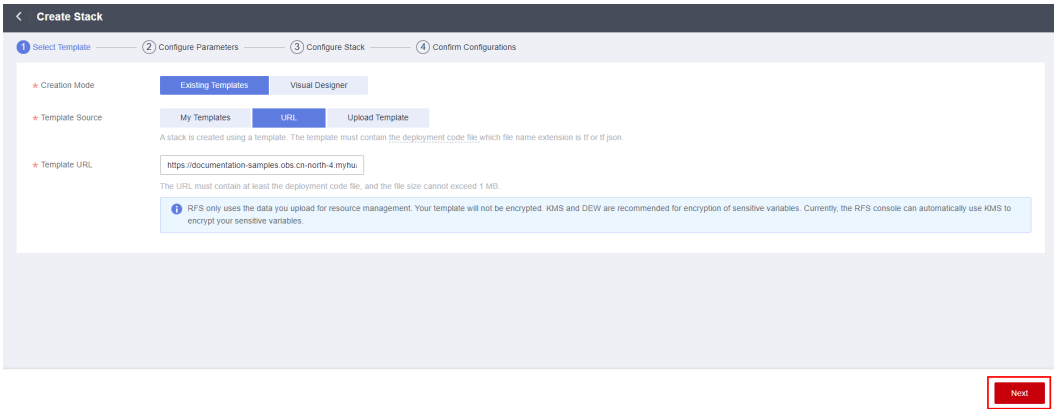
Step 1 Access the *Quick-Start Guides* of Huawei Cloud and choose **Deploying a Scalable HPC Cluster with Slurm**.

Figure 3-11 Selecting a solution



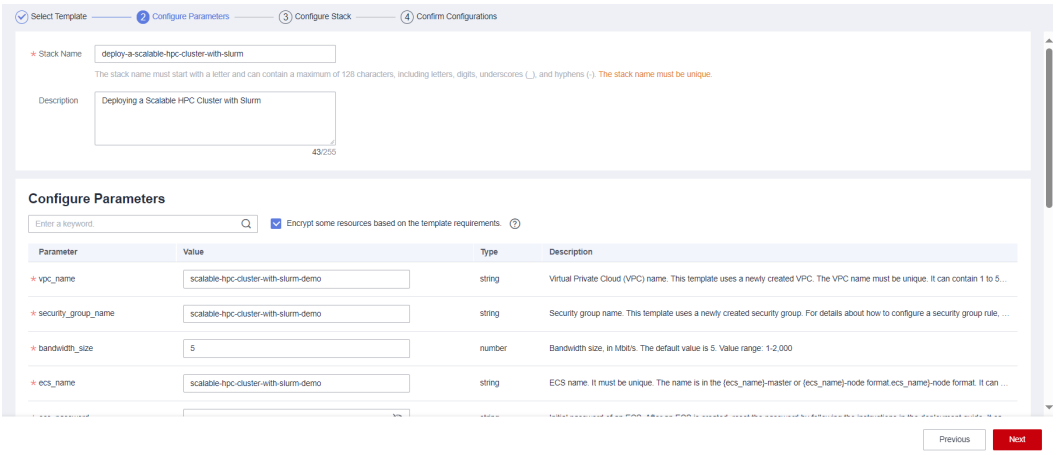
Step 2 Click **Deploy** to switch to the **Create Stack** page.

Figure 3-12 Easy deployment



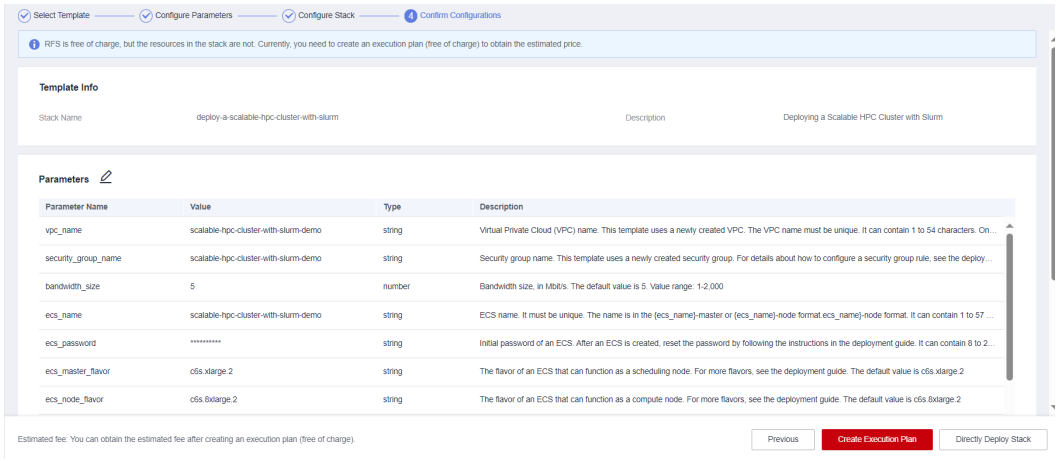
Step 3 Click **Next** and configure user-defined parameters based on **Table 3-1**.

Figure 3-13 Configuring parameters



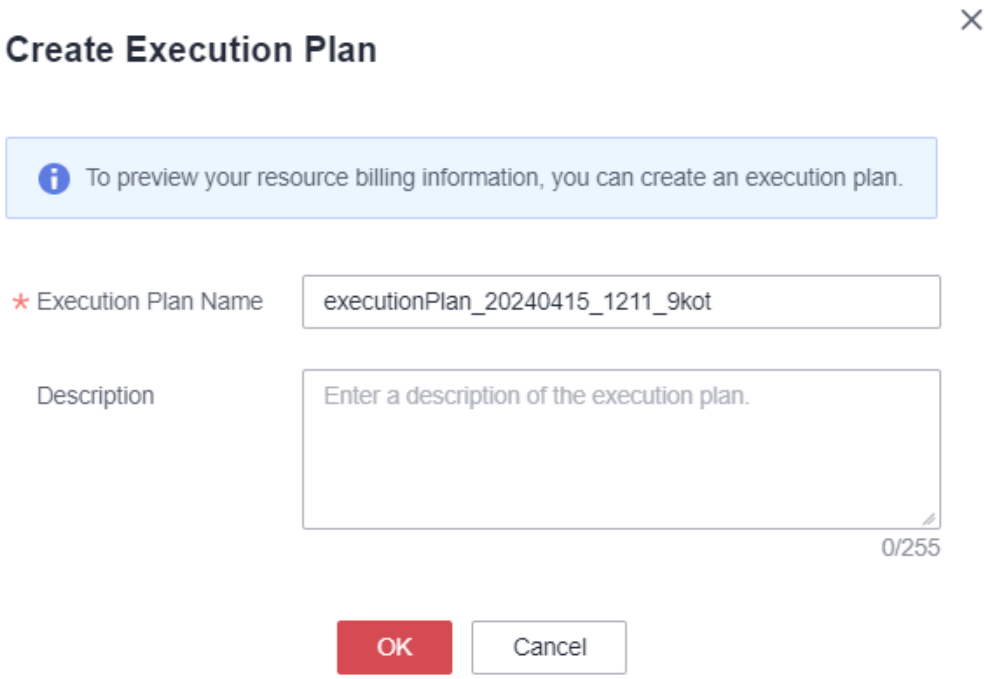
Step 4 Click **Next** to create an execution plan.

Figure 3-14 Creating an execution plan



Step 5 On the **Confirm Configurations** page, click **Create Execution Plan**. In the displayed dialog box, enter an execution plan name and click **OK**.

Figure 3-15 Creating an execution plan



Step 6 Wait until the execution plan is successfully created, and locate that plan and click **Deploy** in the **Operation** column. In the displayed dialog box, click **Execute**.

Figure 3-16 Deploying the execution plan

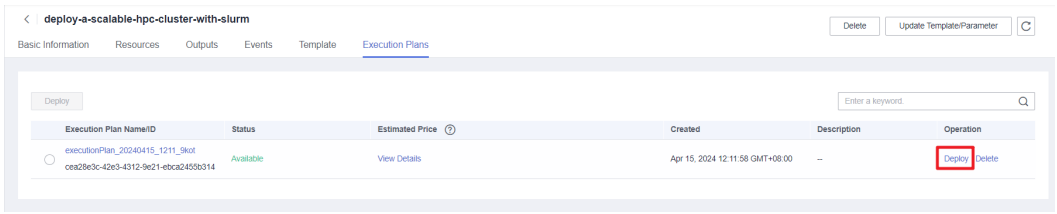
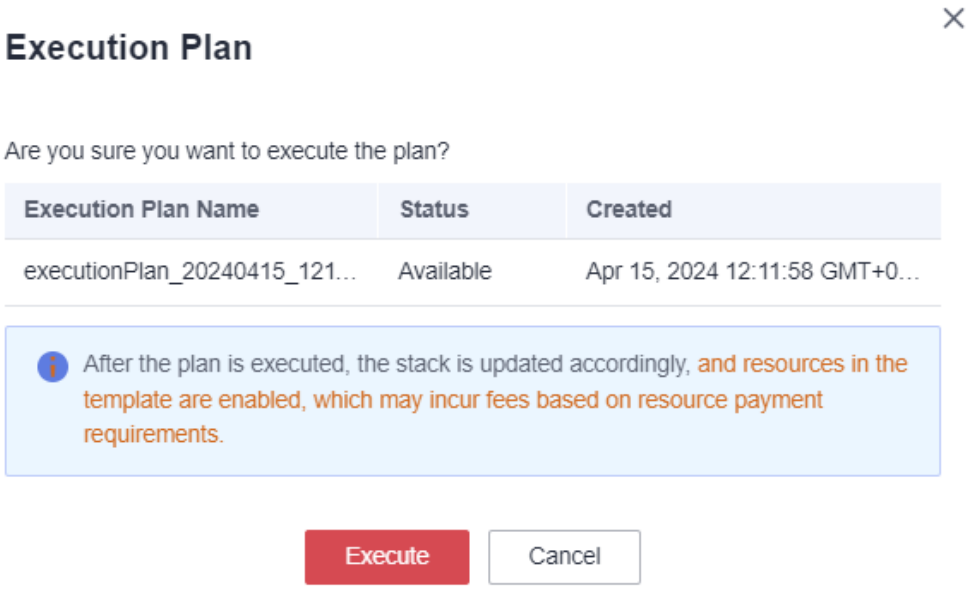


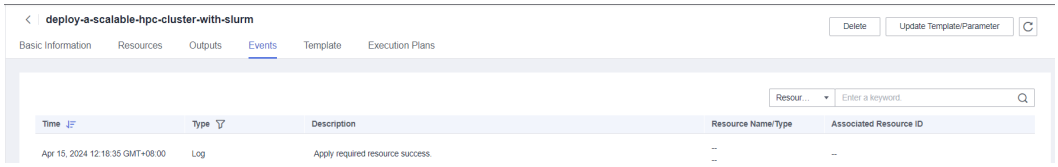
Figure 3-17 Confirming the execution plan



Step 7 (Optional) If you select the yearly/monthly billing mode and your account balance is insufficient, log in to the **Billing Center** to manually pay for the order. You can refer to **Table 2-2** to see the total price.

Step 8 Wait until the execution plan is successfully deployed, as shown in the following figure. Wait for another 30 minutes until the cluster is successfully deployed.

Figure 3-18 Execution plan deployed



Note: Do not log in to any ECS to perform other operations before the cluster deployment is complete. Otherwise, the cluster deployment may fail.

----End

3.3 Getting Started

This section describes the basic usage of Slurm. For the detailed usage, see **Slurm Documentation**.

By default, all service ports involved in this solution are accessible within the same subnet. You can modify the security group rules if needed. The following Slurm service ports are involved:

- 6817: port on the service plane for communicating with Slurm on the scheduling node
- 6818: port on the service plane for communicating with Slurm on compute nodes

NOTE

- By default, the Slurm version 22.05.3 and the Java version jdk1.8.0_202 are used in this solution.
- By default, a cluster named **cluster** has been created in Slurm.

(Optional) Modifying Security Groups

A security group is a collection of access control rules to control traffic to and from cloud resources, such as cloud servers, containers, and databases. Cloud resources associated with the same security group have the same security requirements and are mutually trusted within a VPC.

You can modify security group rules, for example, by adding, modifying, or deleting a TCP port, as follows:

- Adding a security group rule: **Add an inbound rule** and enable a TCP port if needed.
- Modifying a security group rule: Inappropriate security group settings may cause serious security risks. You can **modify a security group rule** to ensure network security of your ECSs.
- Deleting a security group rule: If the source or destination IP address of an inbound or outbound security group rule changes, or a port does not need to be enabled, you can **delete the security group rule**.

Checking Node Status

Step 1 Check the Slurm status on the scheduling node. Specifically, run the **systemctl status munge** and **systemctl status slurmctld** commands to check whether Munge and Slurmctld are running properly.

Figure 3-19 Munge running properly

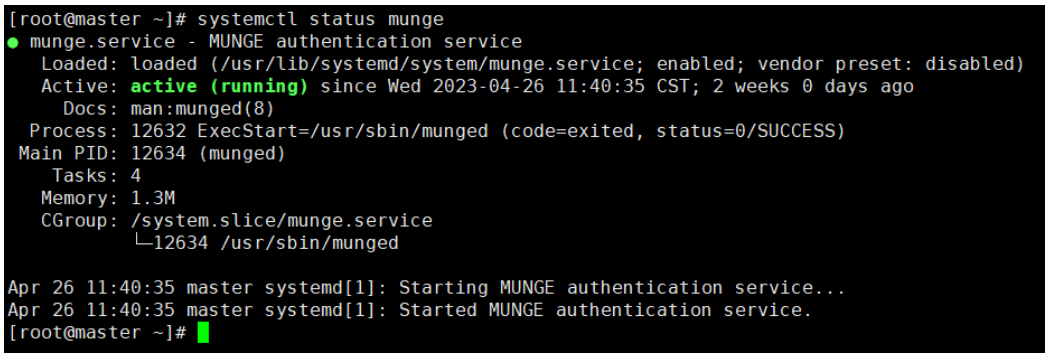
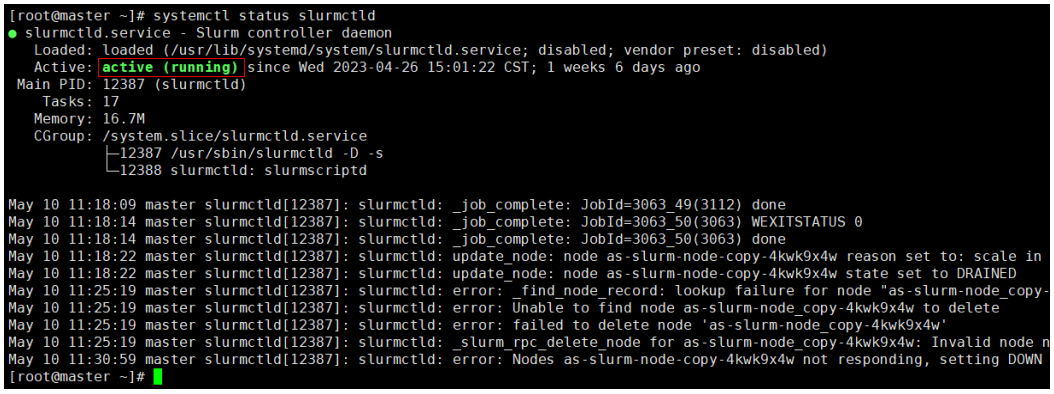


Figure 3-20 Slurmctld running properly



If Munge is not started, run the following command to enable it:

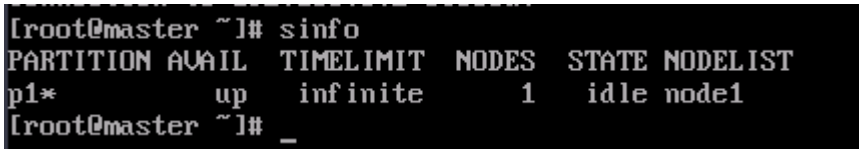
```
systemctl start munge
```

If Slurmctld is not started, run the following command to enable it:

```
systemctl start slurmctld
#Check the node status in the cluster
sinfo
```

Step 2 Each time the cloud server that functions as the scheduling node restarts, enter the username **root** and its password to log in to the cloud server and run the preceding commands to start the services. Run the **sinfo** command to view the cluster node information.

Figure 3-21 Slurm successfully configured



If there are no nodes under **NODELIST**, log in to the compute node to enable Slurmd by running the following command:

```
# Enable Slurmd on node1
ssh 192.168.0.2
systemctl start slurmd
exit
```

Step 3 Check the operating status of Slurm and Gearbox on the scheduling node.

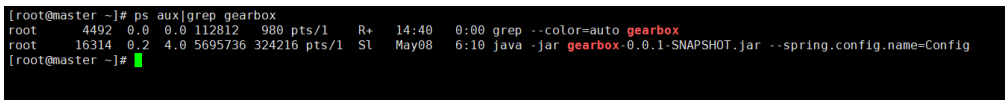
Go to the directory where the Gearbox program package is located (**/usr/local/** by default) and start Gearbox.

```
cd /usr/local/
nohup java -jar gearbox-0.0.1-SNAPSHOT.jar --spring.config.name=gearboxConfig > /dev/null 2>&1 &
```

Step 4 Check the Gearbox status on the scheduling node by running the **ps aux|grep gearbox** command.

```
ps aux|grep gearbox
```

Figure 3-22 Gearbox running properly



Step 5 Set the quota for the Slurm root user. This quota is related to the number of ECSs for auto scaling.

```
#Modify the quota
sacctmgr modify user root set GrpTRES="node={required quota}"
#List the quota
sacctmgr list ass
```

Note: After Gearbox is started, the nodes that do not have computing tasks will be set to the **drain** state. If you want to use the nodes, activate them again by running the following command:

```
scontrol update NodeName={Node name under NODELIST} State=idle
```

Figure 3-23 Node in the drain state

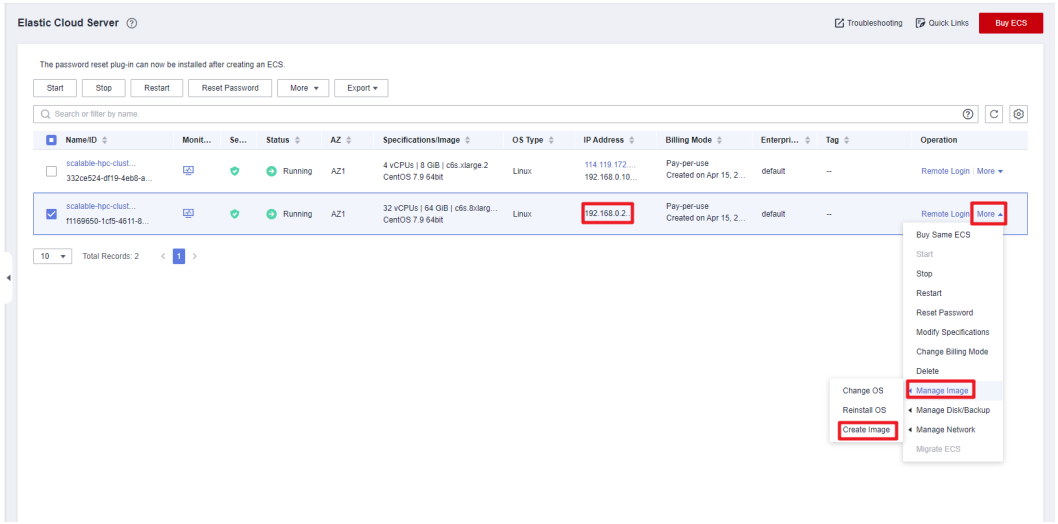
----End

Creating an Image

Step 1 Confirm that the cluster is normal. Then, log in to the **ECS console**.

Step 2 Locate the compute node whose private IP address is 192.168.0.2 and choose **More** > **Manage Image** > **Create Image** in the **Operation** column.

Figure 3-24 Creating an image



Step 3 Set **Image Type** to **System disk image** and set the name to **hpc-image** (an example name).

Figure 3-25 Selecting the system disk image

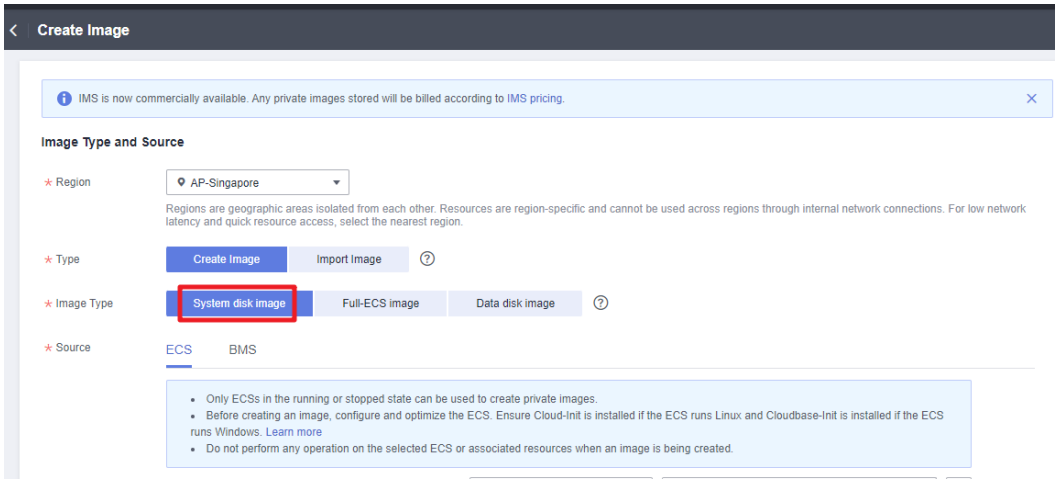
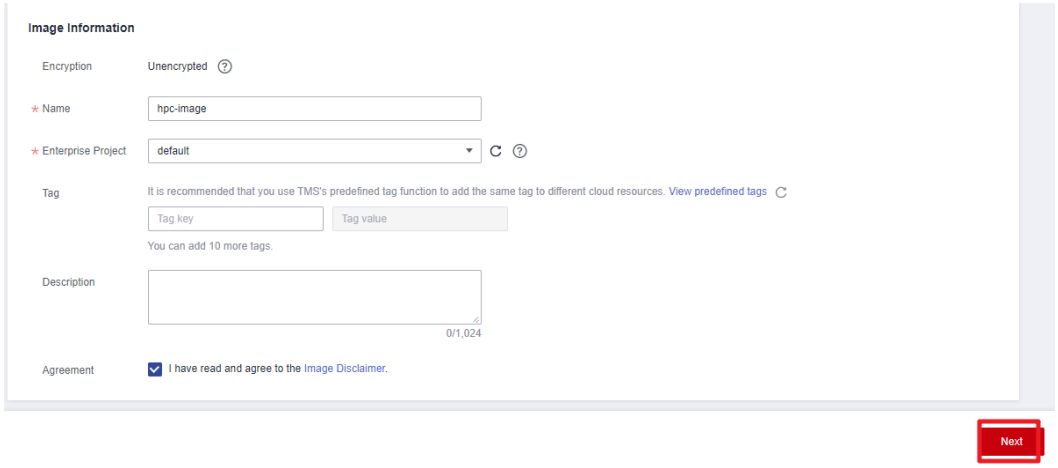


Figure 3-26 Configuring details



Step 4 Click **Next**. You will be navigated to the **IMS console** and can view the image creation progress.

Figure 3-27 An image being created

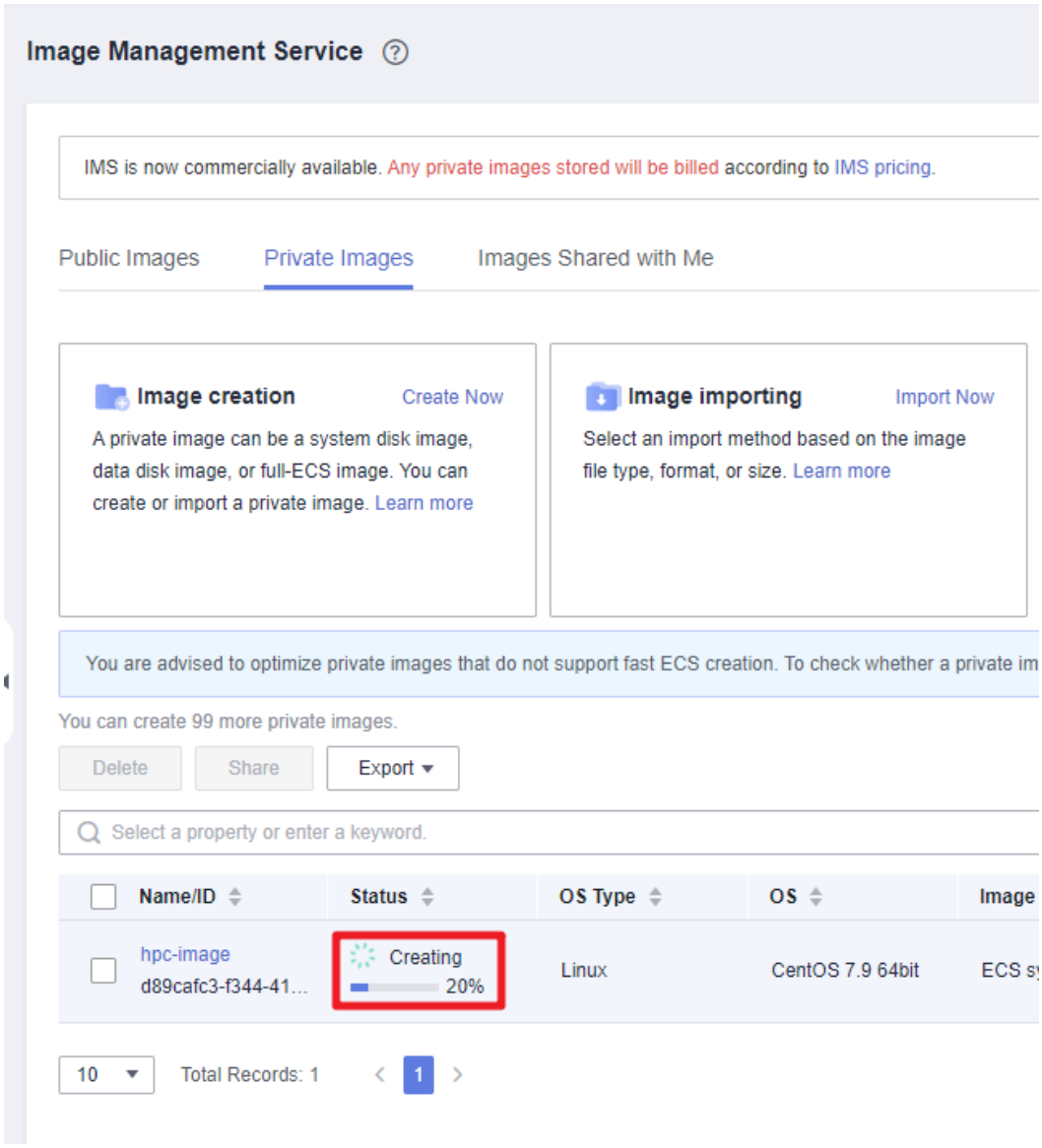
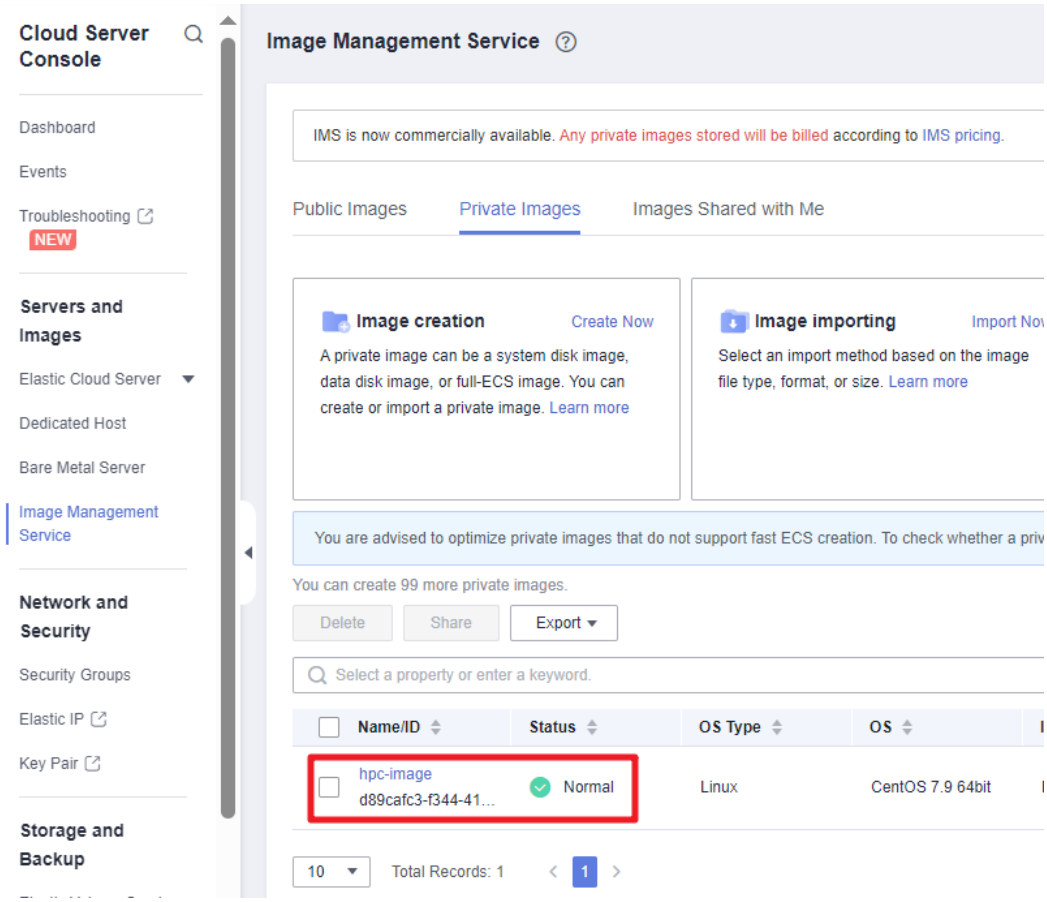


Figure 3-28 An image created



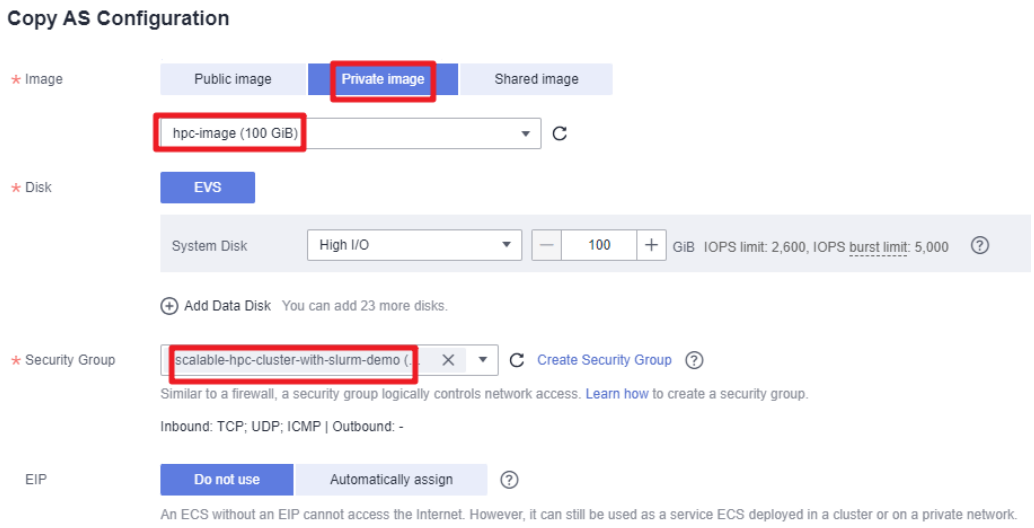
----End

Configuring Auto Scaling

Step 1 Access the **IMS console** to check that your image is successfully created.

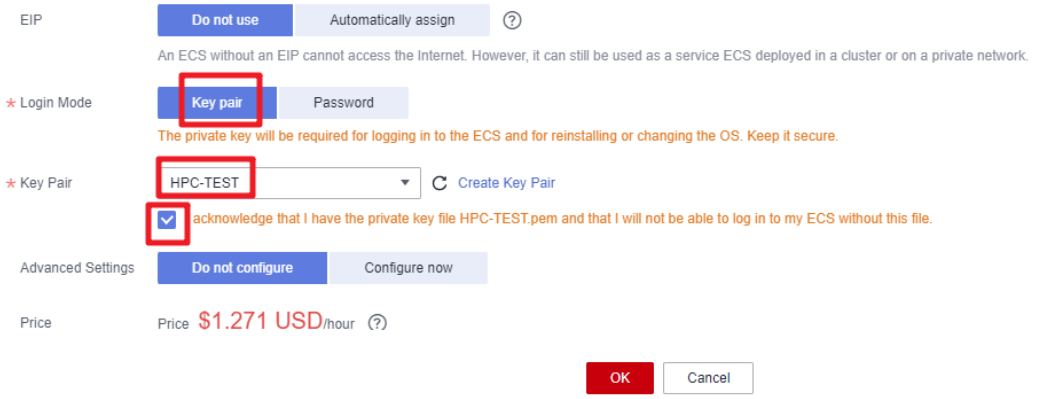
Step 2 Access the Auto Scaling console. Then, locate the **auto scaling configuration** for this solution and click **Copy** in the **Operation** column. In the displayed dialog box, select the private image you created in **Creating an Image** and rename it. Do not include underscores (_) in the name. If underscores are included, instances will not be automatically deleted. Then, click **OK** to create an auto scaling configuration.

Figure 3-29 Creating an auto scaling configuration



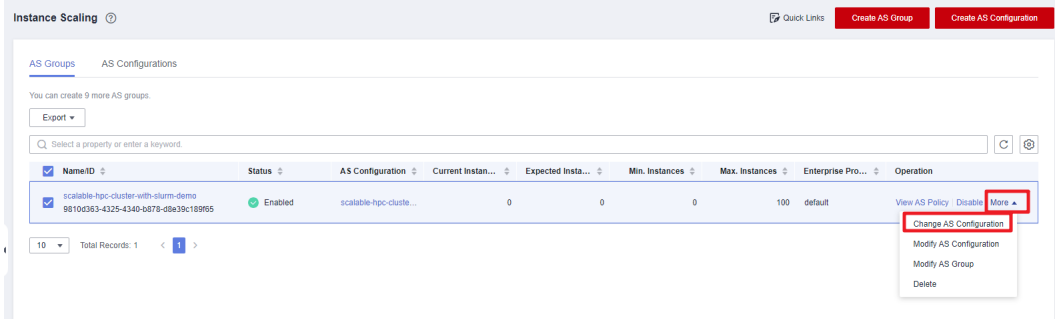
Step 3 Select **Key pair** as the login mode. In this example, set the key pair name to the one (**HPC-TEST** as an example) you specified when you make preparations, and select the check box below.

Figure 3-30 Setting the login mode



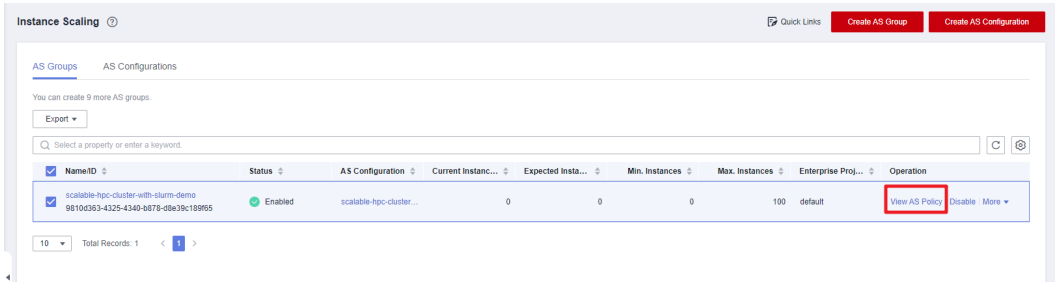
Step 4 In the **auto scaling group list**, locate the group for this solution and choose **More** > **Change AS Configuration** in the **Operation** column. In the displayed dialog box, select the auto scaling c configuration you created and click **OK**.

Figure 3-31 Changing auto scaling configurations



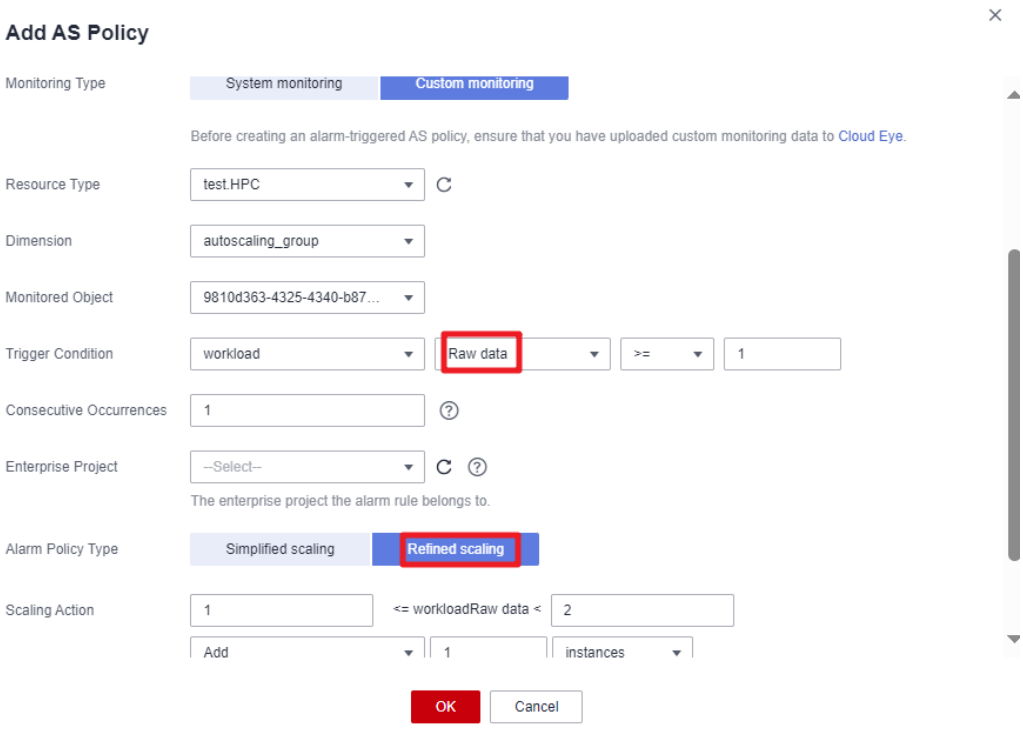
Step 5 On the Huawei Cloud Auto Scaling console, locate the auto scaling group created for this solution and click **View AS Policy** in the **Operation** column.

Figure 3-32 Auto scaling instances



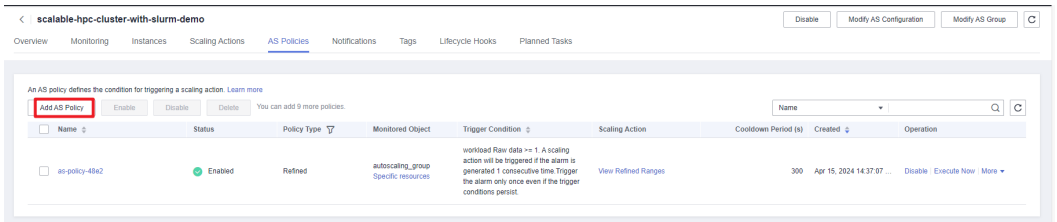
Step 6 Specifically, click **Add AS Policy** and set **Policy Type** to **Alarm**. You can configure up to 10 refined policies for each auto scaling policy if needed. After the configuration is complete, click **OK**.

Figure 3-33 Adding an auto scaling policy



Step 7 Click **Add AS Policy** to add more policies if the number of servers you added for refined policies is less than required.

Figure 3-34 Adding more policies



Step 8 Log in to the ECS and install the required software in the shared file directory **/data**.

Slurm management systems:

- Log file directory for Slurmd, Slurmdbd, and Slurmctld: **/var/log**
- Configuration file for Slurmdbd: **/etc/slurm/slurmdbd.conf**
- Configuration file for Slurmctld and Slurmd: **/etc/slurm/slurm.conf**

  For details, see **Slurm Documentation**. The common commands are as follows:

  Table 3-2 Common commands

Command	Function	Example
sinfo	Shows the system resource usage.	sinfo
squeue	Shows the job status.	squeue
srun	Submits an interactive job.	srun -N 2 -n 48 -p debug A.exe
sbatch	Submits batch jobs.	sbacth -n -2 -n 48 job.sh
salloc	Submits a non-interactive job.	salloc -p debug
scancel	Cancels a job that has been submitted.	scancel JOBID
scontrol	Queries nodes or running jobs.	scontrol show job JOBID
sacct	Queries historical jobs.	sacct -u pp100 -S 03/01/17 -E 03/31/17 --field=joid,partitionjobname,user,nnodes,start,end,elapsed,state

----End

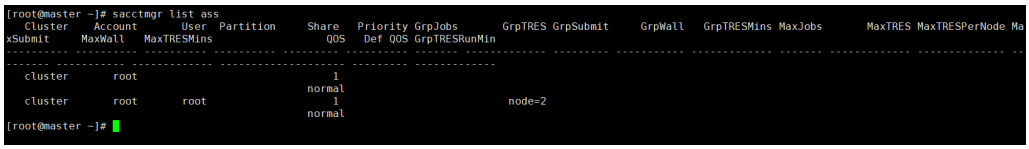
Gearbox Working Process

This section uses batch jobs as an example to describe how Gearbox automatically scales a Slurm cluster.

Step 1 Start Gearbox and set the quota for the Slurm root user.

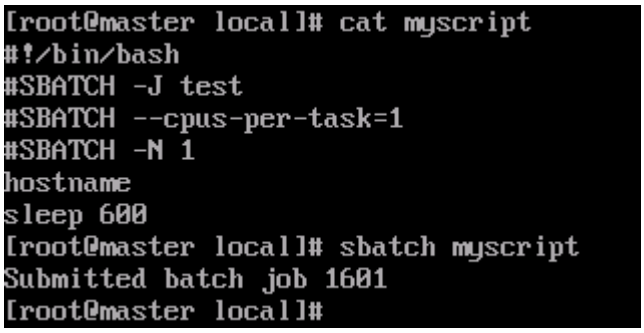
```
#Modify the quota
sacctmgr modify user root set GrpTRES="node={required quota}"
#List the quota
sacctmgr list ass
```


Figure 3-35 Listing the user quota



Step 2 Submit batch jobs (for reference only).

Figure 3-36 Submitting batch slurm jobs

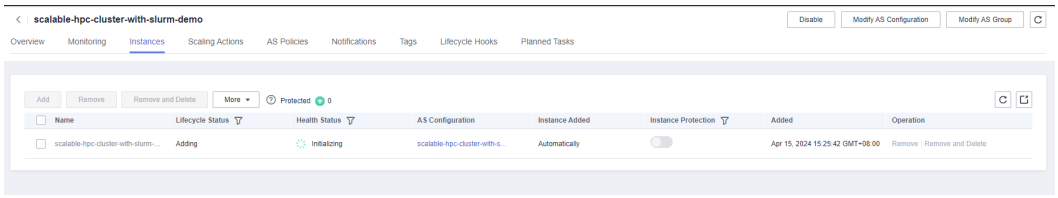


Step 3 View alarm status on the **Cloud Eye console**. If the job status is **queued**, the job is in queue, indicating that the compute node resources in the current cluster are insufficient. If the workload value of custom metrics is greater than or equal to 1, auto scaling instances will be notified to perform auto scale-out.

Figure 3-37 Changes in the workload value

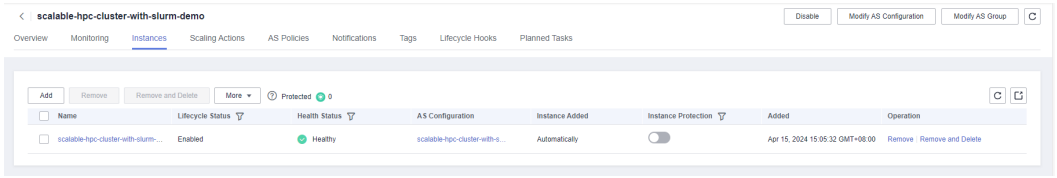
Step 4 Access the **Auto Scaling console**, and click the auto scaling group created for this solution. You will see the instance is being added.

Figure 3-38 Triggering auto scaling



Step 5 Use the image of a compute node for the newly added ECS, log in to the ECS by using the key pair created in **Step 2**, and run the Slurmd program. The ECS will be managed by the Slurm cluster and participate in cluster jobs.

Figure 3-39 ECS successfully added



The Slurm scheduling node distributes job tasks to the compute node added to the Slurm cluster.

Figure 3-40 Executing jobs (added ECSs)

```
Every 1.0s: squeue

             JOBID PARTITION     NAME     USER ST       TIME  NODES NODELIST(REASON)
      3113_[31-50]        p1       ty     root PD       0:00      1 (Resources)
          3113_29        p1       ty     root  R       0:02      1 as-slurm-node-copy-esiu6yqp
          3113_30        p1       ty     root  R       0:02      1 master
```

Step 6 Gearbox periodically checks the status of compute nodes in the Slurm cluster. If there are idle compute nodes, their states will be set to **drain**. Gearbox then checks whether there are jobs running on these compute nodes. If there are not, the compute nodes will be deleted and their states will be set to **down**. Gearbox then identifies all the computes nodes in the **down** or **drain** state and delete them from the Slurm cluster.

Note: **drain** indicates that the node is faulty, **alloc** indicates that the node is in use, **idle** indicates that the node is available, **down** indicates that the node is taken offline, and **mix** indicates that the node is partially occupied with remaining resources available.

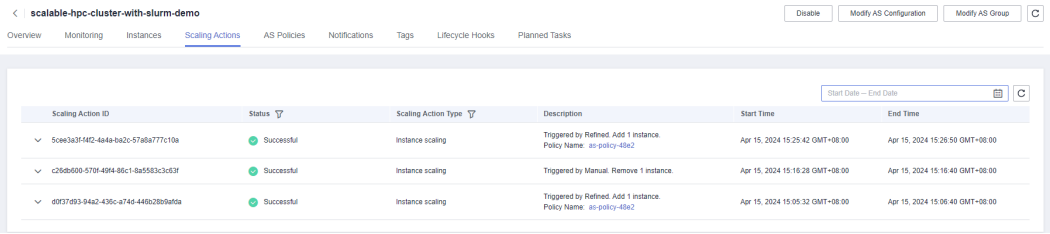
Figure 3-41 Changing the state of an idle instance to drain

```
Every 1.0s: sinfo

PARTITION AVAIL  TIMELIMIT  NODES  STATE NODELIST
p1           up   infinite      1  drain as-slurm-node-copy-esiu6yqp
p1           up   infinite      1   idle master
p2*          up   infinite      0    n/a
```

Step 7 Access the **Auto Scaling console** and click the auto scaling group created for this solution. You will see the instance is being deleted.

Figure 3-42 An idle instance being deleted



----End

Example Gearbox Configuration File

The following is an example Gearbox configuration file. Store it in the Gearbox installation directory. When configuring the **gearboxconfig.yaml** file, delete all comments marked by the number sign (#) in the example.

```
user:
  # AK for the login account
  ak:
  # SK for the login account
  sk:
  # Project ID of the region where the user is located
  project-id:
  # Proxy address, port, username, and password. This parameter is not required if no proxy is needed.
  proxy-address:
  proxy-port:
  proxy-user-name:
  proxy-password:
as:
  # Endpoint domain name of the onsite Auto Scaling
  endpoint: as.ap-southeast-3.myhuaweicloud.com
  # Scaling group ID of the preset scaling group resource
  group:
  # Maximum number of instances displayed on a page. The default value is 100 and cannot be changed.
  list-instance-limit: 100
  # Maximum number of instances to be deleted. The Auto Scaling service supports a maximum of 50 instances. You do not need to change the value.
  delete-instance-limit: 50
ecs:
  # Endpoint domain name of the onsite ECS
  endpoint: ecs.ap-southeast-3.myhuaweicloud.com
metric:
  # Namespace of a custom metric. You do not need to change the value.
  namespace: test.HPC
  # Name of a custom metric
  name: workload
  # Dimension name of a custom metric. You do not need to change the value.
  dimension-name: autoscaling_group
  # Dimension ID of the custom metric. It can be set to the scaling group ID. This value does not affect functions.
  dimension-id:
  #TTL reported by the metric. You do not need to change the value.
  report-ttl: 172800
  # Endpoint domain name of the onsite Cloud Eye
  metric-report-endpoint: ces.ap-southeast-3.myhuaweicloud.com
task:
  # Period for checking the node status, in seconds
  health-audit-period: 30
  # Period for reporting custom metrics, in seconds
  metric-report-period: 60
  # Period for checking whether scale-in is required, in seconds
  scale-in-period: 5
  # Period for automatically deleting nodes to be scaled in, in seconds
  delete-instance-period: 5
```



```
  # Auto-discovery period for newly added nodes, in seconds
  discover-instance-period: 20
  # Period for comparing the number of nodes in the scaling group with that of nodes in the slurm cluster, in seconds
  diff-instance-and-node-period: 60
  # Period for refreshing the internal cache, in seconds
  refresh-cache-period: 100
system:
  # deadline/slurm
  type: slurm
  # Names of stable nodes. Separate multiple names with commas (,).
  stable-nodes: master
  # Partition where stable nodes are located
  stable-partition: p1
  # Partition where variable nodes are located
  variable-partition: p1
  # Period of idle time after which idle nodes will be scaled in, in seconds
  scale-in-time: 1
  # Job waiting time. If the waiting time of a job is longer than this value, the job is considered to be in a queue and will be counted in related metrics. The recommended value is 0.
  job-wait-time: 1
  # Timeout interval for registering a new node, in minutes. If a new node fails to be registered after the timeout interval expires, the node will be deleted in Auto Scaling. The recommended value is 10.
  register-timeout-minutes: 10
  # Number of CPU cores of the elastic node
  cpu: 4
  # Memory size of the elastic node. This is a reserved field. It can be set to any value greater than 0.
  memory: 12600
  # This parameter is valid only for the slurm cluster. The default value is true, indicating that the CPU requirements of jobs with GPU requirements are ignored during workload calculation.
  ignore-cpu-request-of-gpu-job: true
  # Whether to use PowerShell to execute commands. This parameter takes effect only in Windows environments.
  use-powershell-executor: true
```

3.4 Quick Uninstallation

Step 1 Log in to the **Resource Formation Service (RFS) console**, locate the target stack, and click **Delete** in the **Operation** column. In the displayed dialog box, enter **Delete** and click **OK** to uninstall the solution.

Figure 3-43 Uninstalling the solution

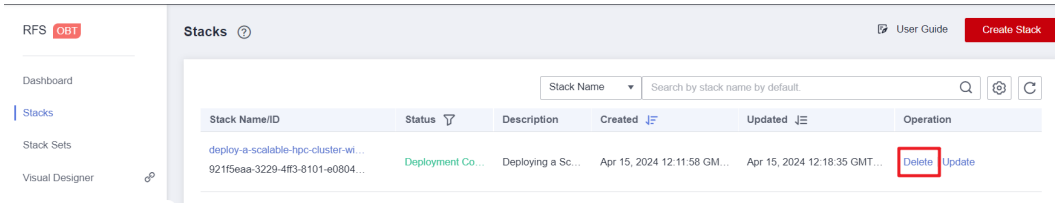
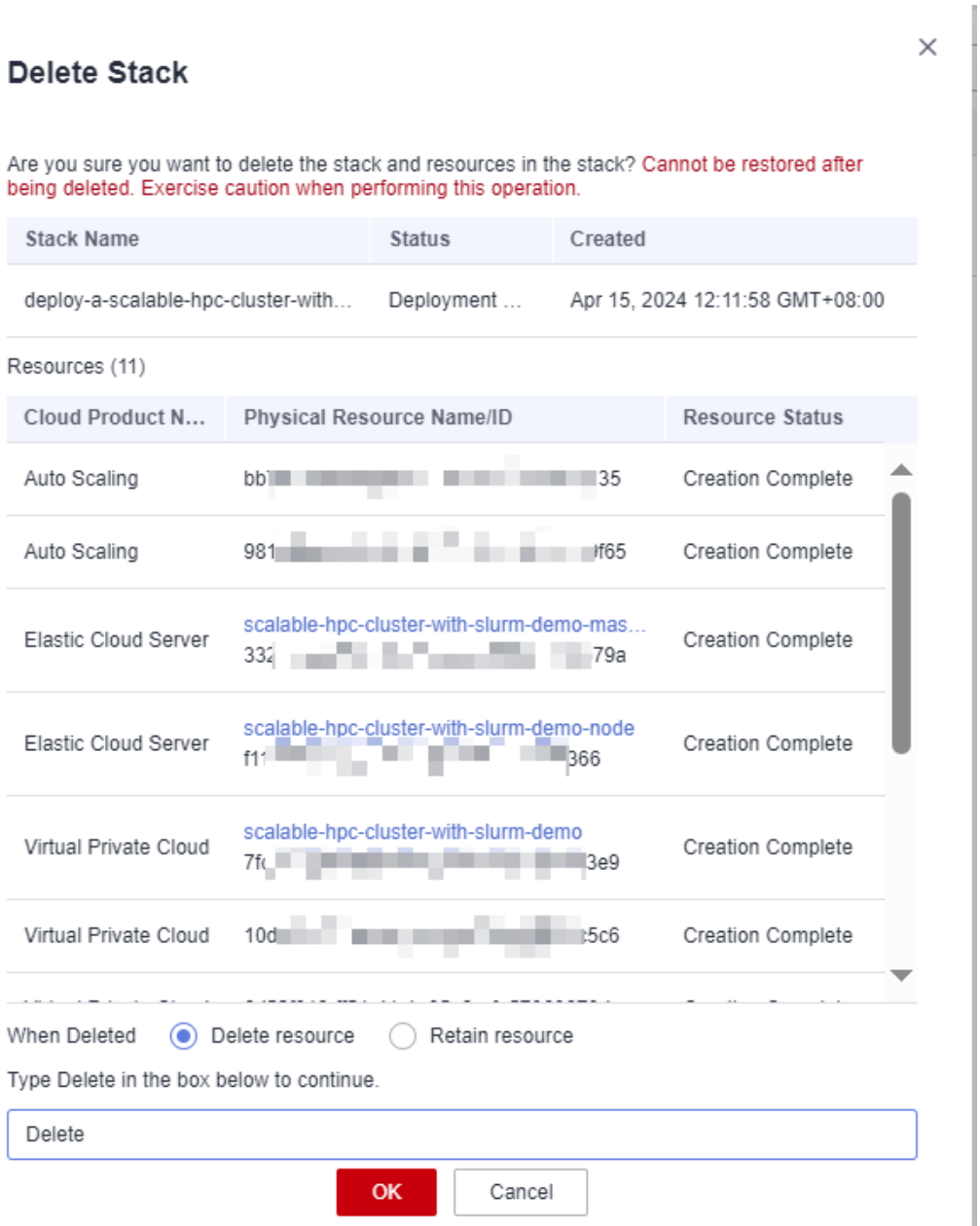
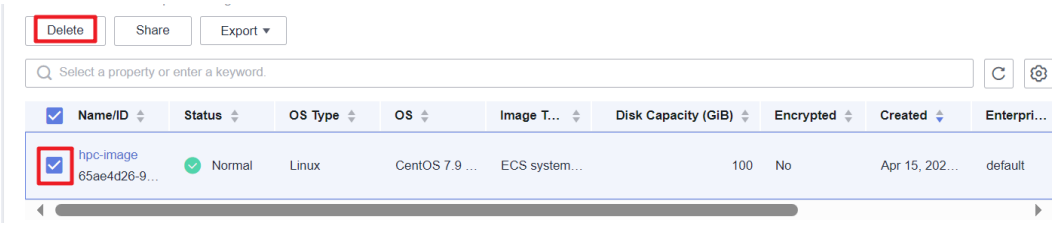


Figure 3-44 Confirming the deletion



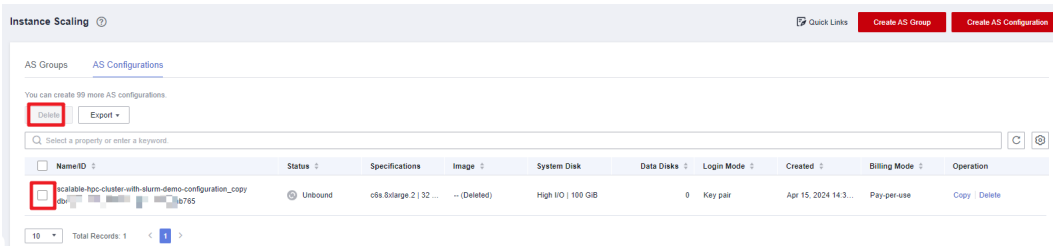
Step 2 Log in to the **IMS console** to manually delete the image you created.

Figure 3-45 Deleting an image



Step 3 Access **auto scaling configurations** to manually delete the auto scaling configuration you created.

Figure 3-46 Deleting an auto scaling configuration



----End

4 Appendix

Terms

- Elastic Cloud Server (ECS): ECS provides secure, scalable, on-demand compute resources, enabling you to flexibly deploy applications and workloads.
- Elastic IP (EIP): EIP enables your cloud resources to communicate with the Internet using static public IP addresses and scalable bandwidth. You can easily bind an EIP to an ECS, BMS, virtual IP address, NAT gateway, or load balancer, enabling immediate Internet access.
- Scalable File Service (SFS) Turbo: SFS Turbo provides a fully hosted shared file storage, which is expandable to 320 TB. It is highly available and stable to support small files and applications requiring low latency and high IOPS.
- Slurm: Slurm is an open-source, highly scalable cluster management tool and job scheduling system for Linux clusters of various scales. It provides the following key features:
  - Resource allocation

    Slurm allocates exclusive or non-exclusive resources of a certain period for users to run jobs.
  - Job management framework

    Slurm provides a framework for starting, executing, and monitoring parallel jobs on the allocated resources.
  - Queue

    Slurm places jobs in a queue when the submitted jobs require more resources than available.
  - Abundant job scheduling policies

    Slurm provides advanced job scheduling policies, such as resource reservation, fair-share scheduling, and backfilling.
  - Other tools

    Slurm provides tools such as job information statistics and job status diagnosis.
- Gearbox: Gearbox is a Huawei's open-source resource collaboration system. It works with, Slurm, a scheduling platform, to coordinate on-premises and cloud resources. Gearbox offers lossless elastic scaling by:

- Supporting custom workload metrics
- Providing metric-based auto scale-out
- Detecting job status
- Releasing idle resources

5 Change History

Table 5-1 Change history

Released On	Description
2023-05-30	This issue is the first official release.
2023-11-30	Revised the solution description and procedure.

