Elastic MapReduce

Getting Started

Product Documentation

Contents

Getting Started
Business Evaluation

Last updated：2023-12-27 09:57:54

Selecting a Cluster Type

Elastic MapReduce (EMR) provides six types of clusters for you to choose from based on your business needs.
Hadoop cluster: Based on open-source Hadoop and the components that form a Hadoop ecosystem, it offers five use cases, including the default use case, ZooKeeper, HBase, Presto, and Kudu, and provides big data solutions for massive data storage, offline/real-time data analysis, streaming data compute, and machine learning.
Druid cluster: Druid is a high-performance real-time analytics database. It supports big data queries in milliseconds and multiple data ingestion methods. It is suitable for real-time big data query scenarios.
ClickHouse cluster: ClickHouse is a column-oriented database management system. It is suitable for data warehouse analysis scenarios such as real-time wide table analysis, real-time BI report analysis, and user behavior analysis.
Doris cluster: Doris is an MPP analytical database product that supports sub-second queries on PB-level, structured data. It is compatible with MySQL protocol and uses the standard SQL syntax. It is suitable for historical report analysis, real-time data analysis, interactive data analysis, etc.
Kafka cluster: Kafka is a distributed, partitioned, multi-replica, and multi-subscriber message processing system based on ZooKeeper coordination. It is suitable for asynchronous processing, message communication, and streaming data receiving and distribution.
StarRocks cluster: StarRocks adopts full vectorization technology. It supports extremely fast and unified OLAP databases. It is suitable for many data analysis scenarios, such as multidimensional, real-time, and high-concurrency analysis.

Selecting Billing Mode

Billing mode for EMR clusters:
Pay-as-you-go: All nodes in a cluster are charged on a pay-as-you-go basis. This is suitable for clusters that exist for a short time or periodically.
Note:
Select the shutdown mode with caution when shutting down a pay-as-you-go EMR cluster node in the CVM console, because EMR nodes do not support the "no charges when shut down" mode.

Selecting Model and Specification

EMR offers a wide variety of CVM models, including EMR Standard, EMR Compute, EMR High IO, EMR MEM Optimized, and EMR Big Data. If you need the CPM model, submit a ticket to us.
You can choose a model based on your business needs and budget.
If you require low latency for offline compute, we recommend you select a model with local disks or the Big Data model.
If you need to use the real-time database HBase, we recommend you select the EMR High IO model with local SSD disks for optimal performance.

Node specification recommendations

EMR offers five types of nodes for your choice based on the cluster type.

Cluster Type	Use Case	Node Type	Recommended Specification
Hadoop	Default use case	Master	Master node: We recommend you select an instance specification with a large memory size (at least 8 GB) and use cloud disks for high stability.
		Core	If most of your data is stored on COS, core nodes will function in a way similar to task nodes and should have a capacity of at least 500 GB. Core nodes cannot be elastically scaled. If your architecture does not use COS, core nodes are responsible for processing cluster compute and storage tasks, and three-replica backup is enabled by default. When estimating the data disk capacity, you need to consider the capacity for storing three replicas. In this case, the Big Data model is recommended.
		Task	If your architecture does not use COS, task nodes are not required. If most of your data is stored on COS, task nodes can be used as elastic compute resources and deployed as needed.
		Common	Common node: It is mainly used as ZooKeeper nodes. You need to select a specification of at least 2 cores, 4 GB memory, and 100 GB cloud disk capacity to meet the requirements.
		Router	Router node: It is mainly used to relieve the load of master nodes and as a task submitter. Therefore, we

			recommend you select a model with a large memory size, preferably not lower than the specification of master nodes.
	ZooKeeper	Common	Common node: It is mainly used as ZooKeeper nodes. You need to select a specification of at least 2 cores, 4 GB memory, and 100 GB cloud disk capacity to meet the requirements.
	HBase	Master	Master node: We recommend you select an instance specification with a large memory size (at least 8 GB) and use cloud disks for high stability.
		Core	If most of your data is stored on COS, core nodes will function in a way similar to task nodes and should have a capacity of at least 500 GB. Note: Core nodes cannot be elastically scaled. If your architecture does not use COS, core nodes are responsible for processing cluster compute and storage tasks.
		Task	If your architecture does not use COS, task nodes are not required. If most of your data is stored on COS, task nodes can be used as elastic compute resources and deployed as needed.
		Common	Common node: It is mainly used as ZooKeeper nodes. You need to select a specification of at least 2 cores, 4 GB memory, and 100 GB cloud disk capacity to meet the requirements.
		Router	Router node: It is mainly used to relieve the load of master nodes and as a task submitter. Therefore, we recommend you select a model with a large memory size, preferably not lower than the specification of master nodes.
	Kudu	Master	Master node: We recommend you select an instance specification with a large memory size (at least 8 GB) and use cloud disks for high stability.
		Core	If most of your data is stored on COS, core nodes will function in a way similar to task nodes and should have a capacity of at least 500 GB. Note: Core nodes cannot be elastically scaled.

			If your architecture does not use COS, core nodes are responsible for processing cluster compute and storage tasks, and three-replica backup is enabled by default. When estimating the data disk capacity, you need to consider the capacity for storing three replicas. In this case, the Big Data model is recommended.
		Task	If your architecture does not use COS, task nodes are not required. If most of your data is stored on COS, task nodes can be used as elastic compute resources and deployed as needed.
		Common	Common node: It is mainly used as ZooKeeper nodes. You need to select a specification of at least 2 cores, 4 GB memory, and 100 GB cloud disk capacity to meet the requirements.
		Router	Router node: It is mainly used to relieve the load of master nodes and as a task submitter. Therefore, we recommend you select a model with a large memory size, preferably not lower than the specification of master nodes.
	Presto	Master	Master node: We recommend you select an instance specification with a large memory size (at least 8 GB) and use cloud disks for high stability.
		Core	If most of your data is stored on COS, core nodes will function in a way similar to task nodes and should have a capacity of at least 500 GB. Note: Core nodes cannot be elastically scaled. If your architecture does not use COS, core nodes are responsible for processing cluster compute and storage tasks, and three-replica backup is enabled by default. When estimating the data disk capacity, you need to consider the capacity for storing three replicas. In this case, the Big Data model is recommended.
		Task	If your architecture does not use COS, task nodes are not required. If most of your data is stored on COS, task nodes can be used as elastic compute resources and deployed as needed.

		Common	Common node: It is mainly used as ZooKeeper nodes. You need to select a specification of at least 2 cores, 4 GB memory, and 100 GB cloud disk capacity to meet the requirements.
		Router	Router node: It is mainly used to relieve the load of master nodes and as a task submitter. Therefore, we recommend you select a model with a large memory size, preferably not lower than the specification of master nodes.
ClickHouse	Default use case	Core	Core node: We recommend you select a model with high CPU and a large memory size. Because data may be lost if a local disk is corrupted, cloud disks are recommended.
		Common	Common node: The CPU and memory configuration should be at least 4 cores and 16 GB.
Kafka	Default use case	Core	Core node: We recommend you select a model with high CPU and a large memory size. Because data may be lost if a local disk is corrupted, cloud disks are recommended.
		Common	Common node: The CPU and memory configuration should be at least 4 cores and 16 GB.
Doris	Default use case	Master	Master node: We recommend you select an instance specification with a large memory size (at least 8 GB) and store all the metadata of master nodes in the memory.
		Core	Core node: We recommend you select an instance specification with a large memory size (at least 8 GB) and use cloud SSD for better IO performance and stability.
		Router	Router node: The frontend module is deployed here for high read/write availability. Therefore, we recommend you select a model with a large memory size, preferably not less than that of master nodes.
Druid	Default use case	Master	Master node: We recommend you select an instance specification with a large memory size (at least 16 GB) and use SSD disks for better IO performance.
		Core	Core node: We recommend you select an instance

			specification with a large memory size (at least 8 GB) and use cloud SSD for better IO performance and stability.
		Task	If your architecture does not use COS, task nodes are not required. If most of your data is stored on COS, task nodes can be used as elastic compute resources and deployed as needed.
		Common	Common node: It is mainly used as ZooKeeper nodes. We recommend you select the specification of 2 cores, 4 GB memory, and 100 GB cloud disk capacity to meet the requirements.
		Router	Router node: It is mainly used to relieve the load of master nodes and as a task submitter. Therefore, we recommend you select a model with a large memory size, preferably not lower than the specification of master nodes.
StarRocks	Default use case	Master	Master node: We recommend you select an instance specification with a large memory size (at least 8 GB) and store all the metadata of master nodes in the memory.
		Core	Core node: We recommend you select an instance specification with a large memory size (at least 8 GB) and use cloud SSD for better IO performance and stability.
		Router	Router node: The frontend module is deployed here for high read/write availability. Therefore, we recommend you select a model with a large memory size, preferably not less than that of master nodes.

Note:

Different cluster types have different requirements for the node specification. Currently, the system automatically recommends the configuration that meets the cluster's requirements by default. You can adjust the model specification based on your business needs, and the recommended model is for reference only.

Core nodes cannot be elastically scaled. If your architecture does not use COS, core nodes are responsible for processing cluster compute and storage tasks, and three-replica backup is enabled by default. When estimating the data disk capacity, you need to consider the capacity for storing three replicas. In this case, the Big Data model is recommended.

Network and Security

To ensure the network security, the EMR cluster is placed in a VPC, and a security group policy is added to the VPC. In addition, to ensure easy access to the WebUI of Hadoop, a public IP is enabled for one of the master nodes and the node is billed by traffic. A public IP is not enabled for router nodes by default. However, you can bind a router node to an EIP in the CVM console to enable a public IP for it.

Note:

A public IP is enabled for master nodes when a cluster is created. You can disable it as needed.

Enabling a public IP for master nodes is mainly for SSH login and component WebUI access.

Master nodes with a public IP enabled are billed by traffic with a bandwidth of up to 5 Mbps. You can adjust the network in the console after creating a cluster.

Creating EMR Cluster

Last updated：2023-12-27 09:58:17

Overview

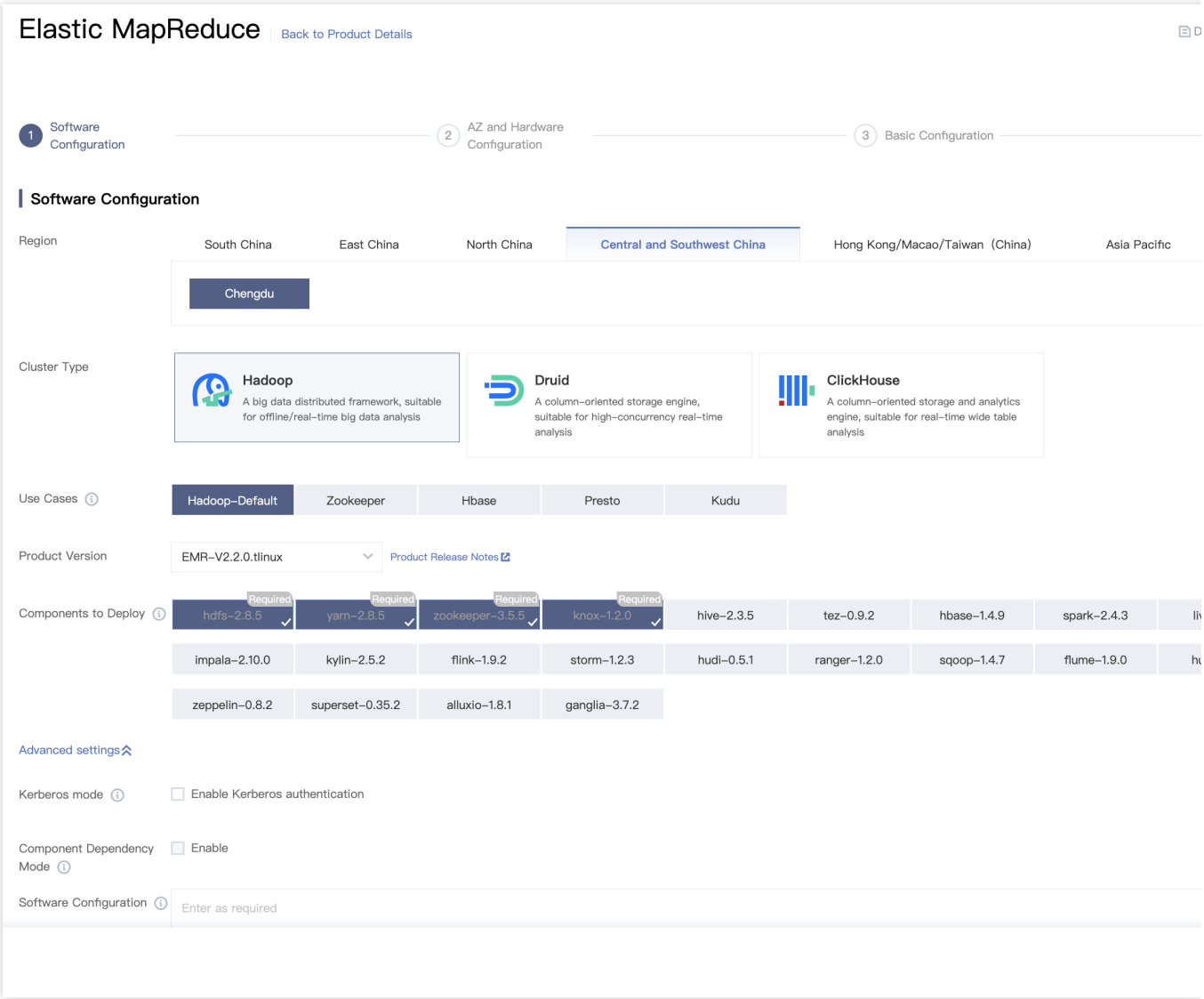
This document describes how to create an EMR cluster in the EMR console.

Directions

Log in to the EMR console and click **Create cluster** on the cluster list page.

1. Software configuration

Region: A region is the physical location of an IDC. Currently supported regions include Guangzhou, Shanghai, Beijing, Singapore, Silicon Valley, Chengdu, Nanjing, and Mumbai. Tencent Cloud products in different regions cannot communicate with each other over a private network.
Cluster type: There are six types of EMR clusters, namely, Hadoop, ClickHouse, Druid, Doris, Kafka, and StarRocks. You can choose one to deploy as needed.
Use cases: Hadoop clusters support five use cases, namely, Hadoop-Default, ZooKeeper, HBase, Presto, and Kudu. You can choose one to deploy as needed.
Product version and **Components to deploy**: EMR recommends some commonly used combinations of components for Hadoop. You can also combine the components based on your needs.
Kerberos mode: It specifies whether to enable Kerberos authentication for the cluster. This feature is not required for individual users and disabled by default.
Software configuration: You can create a cluster by entering custom parameters as required. The external cluster access feature is provided as well, so that you can read/write external cluster data after configuring the correct address information in relevant parameters.



2. AZ and hardware configuration

Billing mode: Pay-as-you-go is supported.

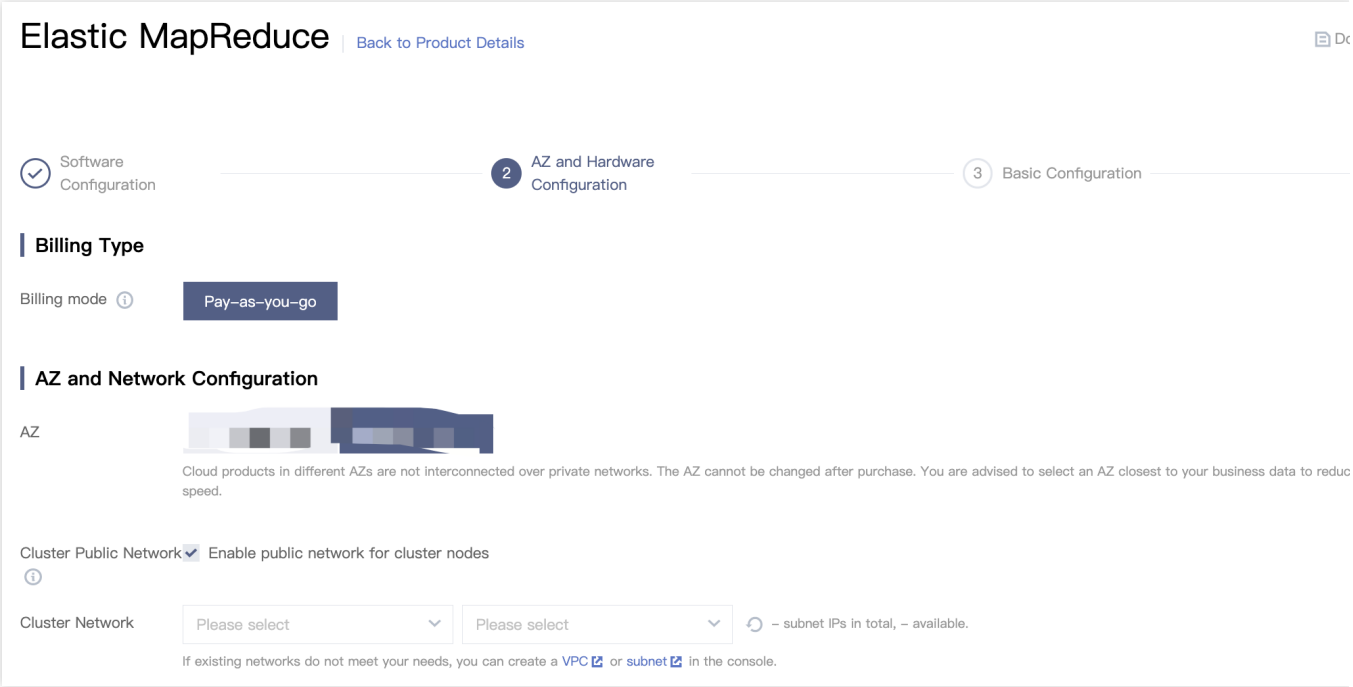
Pay-as-you-go: You are charged by usage duration of the cluster. This billing mode requires identity verification and the amount of two hours' usage fees will be frozen when the cluster is created (vouchers cannot be used here). After this cluster is terminated, the frozen amount will be refunded.

AZ: Different AZs in the same region support different models and specifications. Tencent Cloud products in different regions cannot communicate with each other over a private network. The AZ cannot be changed after purchase. We recommend you select a new AZ closest to the region of your business data to reduce the access latency and increase the download speed.

Cluster network: To ensure the security of the EMR cluster, all nodes of the cluster are placed in a VPC; therefore, you need to set up a VPC before creating the EMR cluster.

Security group: The security group has a firewall feature and is used to set the network access control of the CVM instance. You can use an existing security group. If there is no existing one, EMR will automatically create one for you. If the number of security groups has reached the upper limit and new ones cannot be created, you can delete some unnecessary ones after checking the security groups that are being used.

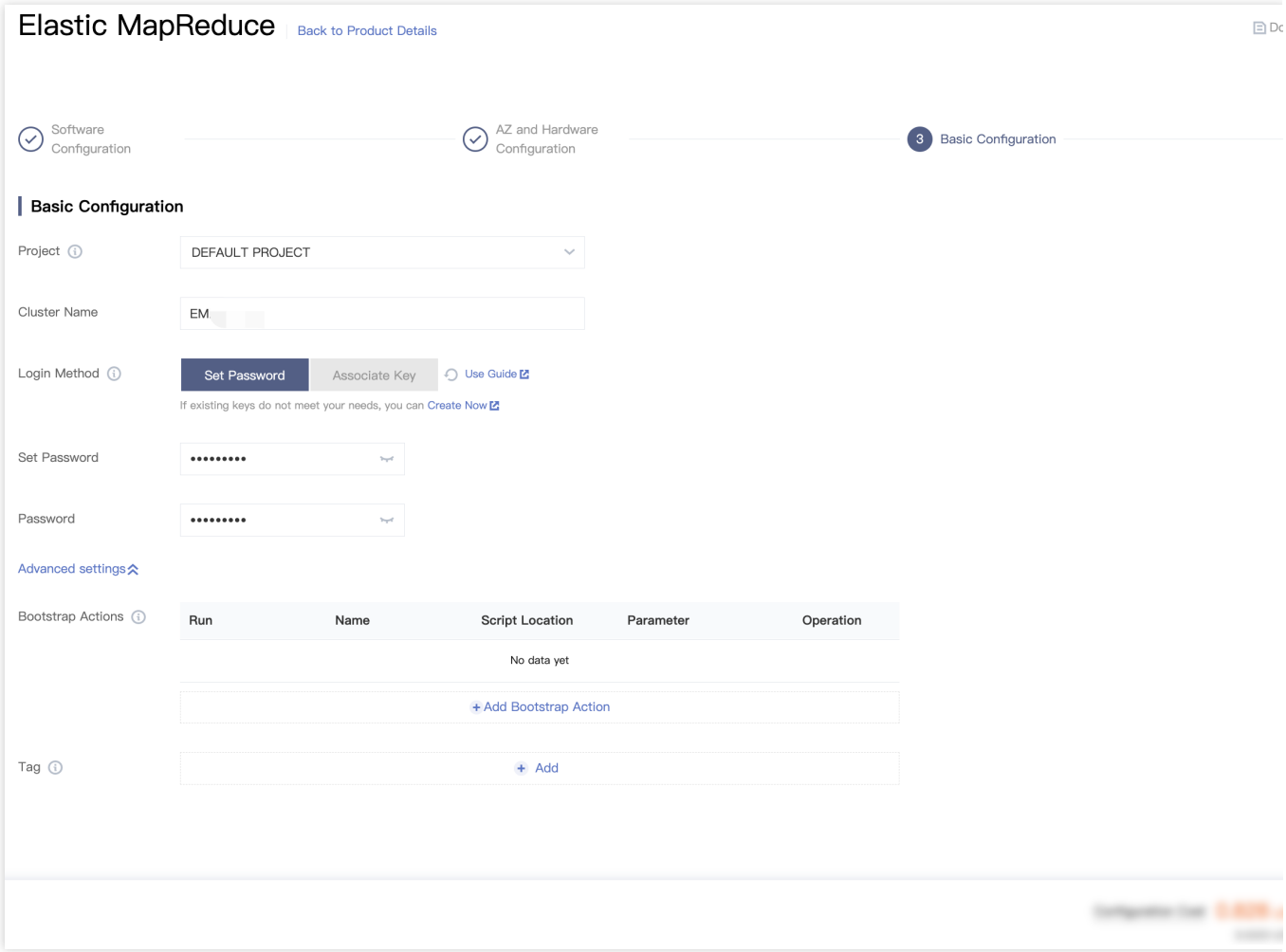
Create a security group: EMR will create a security group that allows traffic going through ports 22 and 30001 as well as all traffic from the necessary private network IP range.
Use an existing EMR security group: Select an existing EMR security group as the security group for your instance. Open ports 22 and 30001 as well as the required IP range for communication over the private network.
Remote login: Port 22 is usually used for remote login and opened on the newly created security group by default. You can close it based on your business needs.
High availability (HA): High availability is enabled by default. The number of different types of nodes deployed varies by cluster type and use case under HA or non-HA mode. For more information, see Cluster Types.
Node configuration: EMR offers multiple node types. You can select an appropriate model configuration for each node type based on your business needs.
Note:
Currently, up to 15 cloud disks of multiple types (one type can be selected only once) can be mounted to a core, task, or router node.



Placement group: It is a policy for distributing and placing CVM instances on the underlying hardware. For more information, see [Placement Group].
Hive metadatabase: If you choose to deploy the Hive component, there are two storage methods for Hive metadata. You can store the metadata in a MetaDB instance separately purchased for the cluster, or associate the metadata with EMR-MetaDB or a self-created MySQL database. In the latter case, metadata will be stored in an associated database and not be cleared when the cluster is terminated.

3. Basic configuration

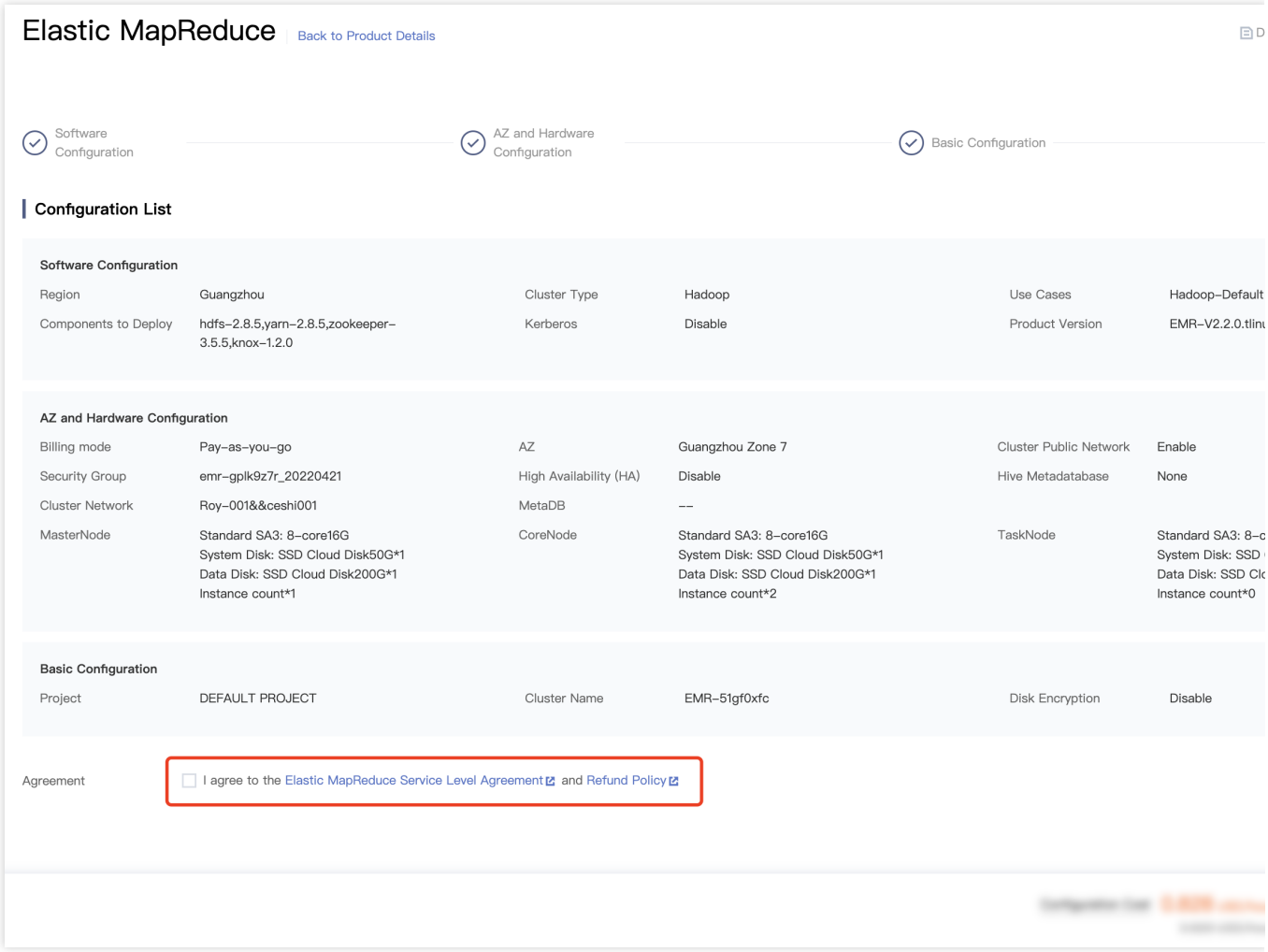
Project: Assign the current cluster to a project. To assign an instance to a new project, create a project first. For detailed directions, see [Creating Project].
Cluster name: Use names to differentiate EMR clusters.
Login method: Currently, EMR provides two ways to log in to cluster services, nodes, and MetaDB, namely, custom password and associated key. SSH keys are only used for logging in to the EMR-UI via the quick entry. The default username is "root", and the username for the WebUI quick entry of the Superset component is "admin".
Advanced settings:
Bootstrap actions: A bootstrap action is a custom script executed when a cluster is created to help you modify the cluster environment, install third-party software, and use your own data.
Tag: You can add tags to clusters or node resources during resource creation to facilitate resource management. Up to 5 tags can be bound to a cluster, and the tag keys must be unique.



4. Configuration confirmation

Auto-renewal: During the seven days before the cluster expires, the system will check whether your account balance is sufficient every day in order to renew the cluster resources with auto-renewal enabled.
After completing the configurations above, click **Purchase** to make the payment. Your EMR cluster will be

automatically created once the payment is received. You will find the cluster you just created in the EMR console after about 10 minutes.



Note:

You can view the information of each node in the CVM console. To ensure your EMR cluster works properly, do not modify such information.

Subsequent operations

After the cluster is successfully created, you can log in to it and further configure and perform other operations on it. For specific operations, see the following documents:

Logging In to Clusters

Cluster configuration: Software Configuration, Mounting CHDFS Instance, and Unified Management of Hive Metadata.

Cluster management: Setting Tag, Bootstrap Actions, and Cluster Termination.

Logging in to Clusters

Last updated：2023-12-27 09:58:34

Logging in Through a Remote Login Tool (on Windows)

This section uses Xshell as an example to describe how to log in to an EMR cluster with a password by using a remote login tool on Windows.

Applicable OS

Windows

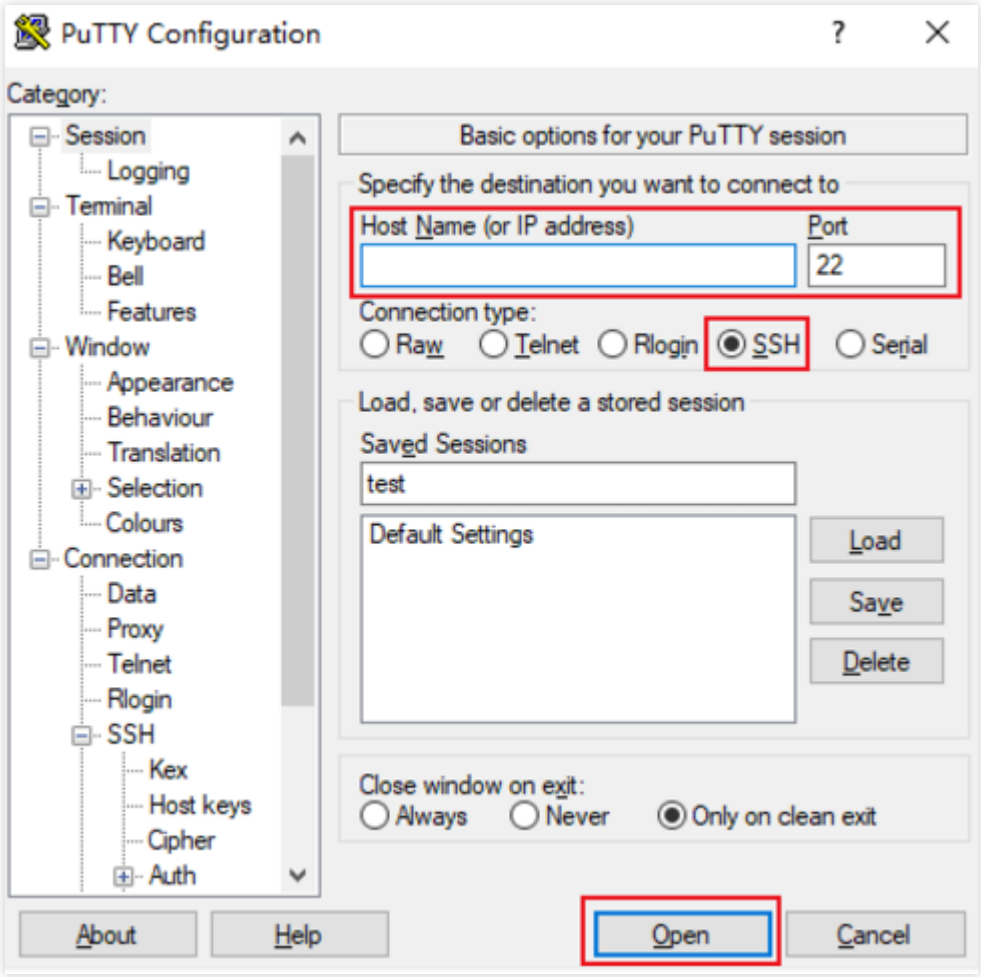
Logging in with a password

1. Download PuTTY (a remote login tool on Windows) here and install it.
2. Launch the PuTTY client, enter the following in the PuTTY Configuration window, and click **Open** to create a session as shown below:

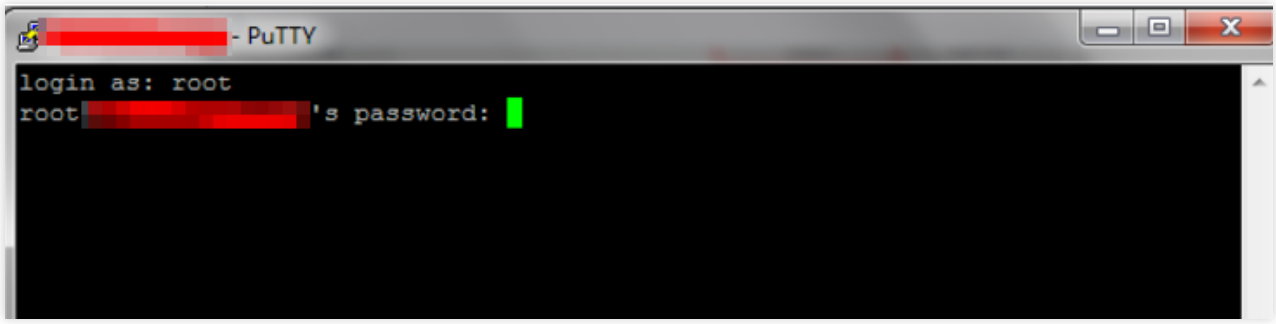
Host Name: The public IP address of the EMR cluster, which can be viewed on the list page or details page in the EMR Console.

Port: The port number of the CVM instance, which has to be 22. Please make sure that the port 22 in the CVM instance is opened. For more information, please see Security Group and Network ACL.

Connection type: Select **SSH**.



3. In the PuTTY session window, enter the obtained admin account and press Enter.
4. Enter the obtained login password and press Enter to log in as shown below:



Log in Through SSH (on Linux or macOS)

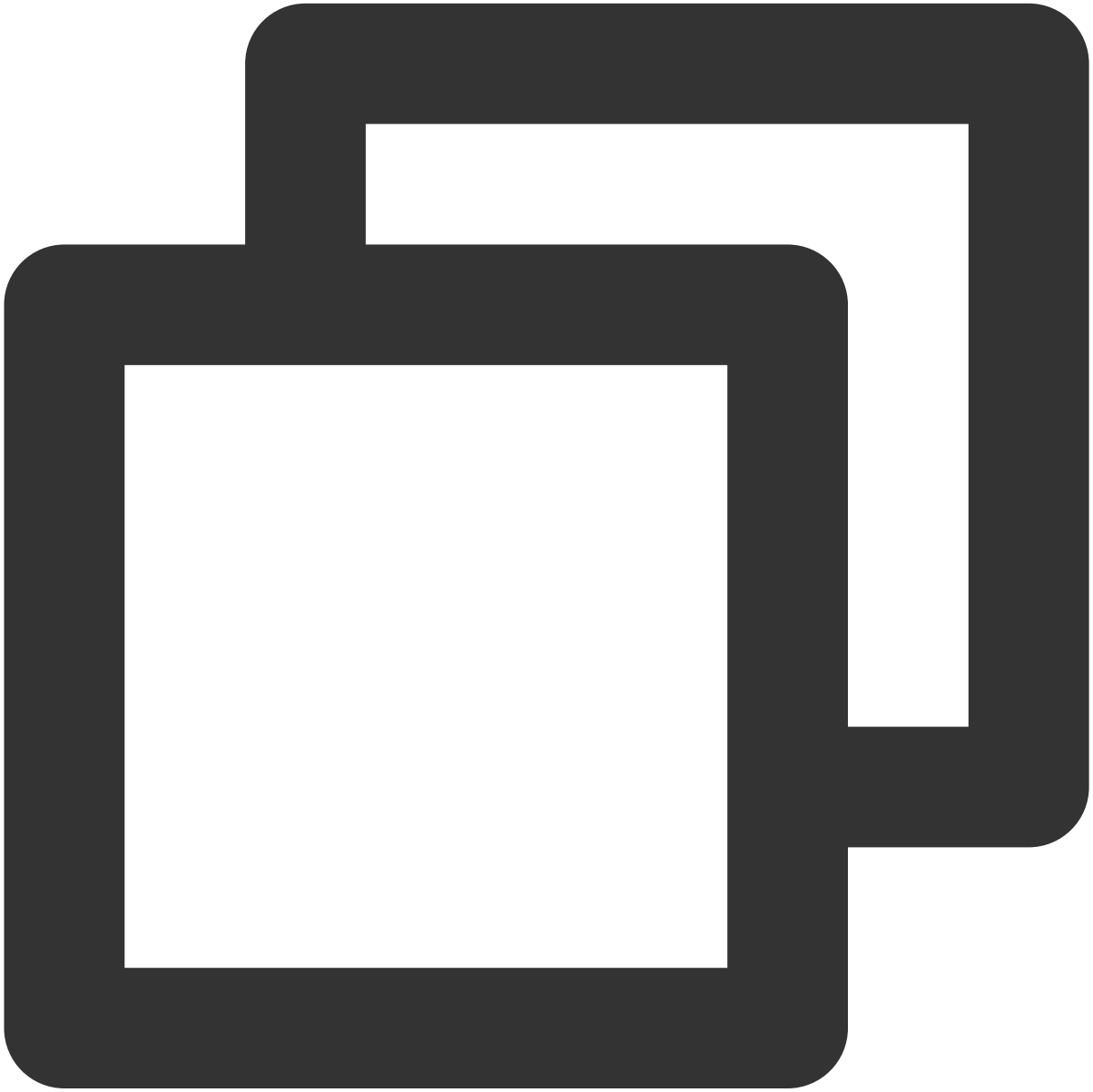
This section describes how to log in to an EMR cluster through SSH on Linux or macOS.

Applicable OS

Linux or Mac OS

Logging in with a password

1. On macOS, launch Terminal and run the following command. On Linux, run the following commands directly:



```
ssh <username>@<hostname or IP address>
```

username: The admin account, such as root.

hostname or IP address: The public IP address or custom domain name of your EMR cluster.

2. Enter the obtained password (only the input but not the output is displayed here) and press Enter to log in.