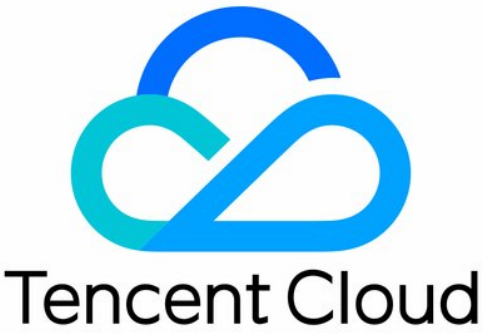
Elastic MapReduce

Best Practices

Product Documentation



Contents

Best Practices
Data Migration
Getting Started

Last updated：2020-03-25 11:00:35

EMR currently allows you to migrate your data in two ways: 1. Migrating your data to COS; 2. Copying EMR cluster data to your self-built HDFS through DistCP. This method requires your self-built cluster is connect with the EMR cluster network.

Migrating Data Through COS

Last updated：2023-02-23 14:17:19

Migrating a non-HDFS file

If your source file is not an HDFS file, upload it to COS via the COS console or API, and then analyze it in the EMR cluster.

Migrating an HDFS file

1. Get the COS migration tool

   Get the migration tool hdfs_to_cos_tools. For more migration tools, see Tool Overview.
2. Configuring the tool

   All configuration files are stored in the `conf` directory of the tool directory. Copy the `core-site.xml` file of the HDFS cluster to be synced to `conf`, which contains the configuration information of the NameNode. Edit the configuration file `cos_info.conf` by including your `appid`, bucket, region, and key information.
3. Command parameter descriptions

> Note：
>
> - We recommend you use a sub-account key and follow the principle of least privilege to avoid leaking resources besides your buckets and objects.
> - If you need to use a permanent key, we recommend you follow the principle of least privilege to limit the scope of permission for the permanent key.

```
-ak <ak> the cos secret id // Your `SecretId`. We recommend you use a sub-account key and follow the principle of least privilege to reduce risks. For information on how to get a sub-account key, visit https://www.tencentcloud.com/document/product/598/32675.
-appid,--appid <appid> the cos appid
-bucket,--bucket <bucket_name> the cos bucket name
-cos_info_file,--cos_info_file <arg> the cos user info config default is ./conf/cos_info.conf
-cos_path,--cos_path <cos_path> the absolute cos folder path
-h,--help print help message
-hdfs_conf_file,--hdfs_conf_file <arg> the hdfs info config default is ./conf/core-site.xml
-hdfs_path,--hdfs_path <hdfs_path> the hdfs path
-region,--region <region> the cos region. legal value cn-south, cn-east, cn-north, sg
```

```
-sk <sk> the cos secret key // Your `SecretKey`. We recommend you use a sub-accou
nt key and follow the principle of least privilege to reduce risks. For informati
on on how to get a sub-account key, visit https://www.tencentcloud.com/document/p
roduct/598/32675.
-skip_if_len_match,--skip_if_len_match skip upload if hadoop file length match co
s
```

4. Executing data migration

```
# All operations must be performed in the tool directory. If both configuration
files and command line parameters are set, the latter will prevail
./hdfs_to_cos_cmd -h
# Copy from HDFS to COS (if a file already exists in COS, it will be overwritte
n)
./hdfs_to_cos_cmd --hdfs_path=/tmp/hive --cos_path=/hdfs/20170224/
# Copy from HDFS to COS, and if a file to be copied is of the same length as a
file in COS, then it is skipped (this is suitable for repeated copy)
# Only the length is checked here, as the overheads would be very high if the d
igests of files in Hadoop are to be calculated
./hdfs_to_cos_cmd --hdfs_path=/tmp/hive --cos_path=/hdfs/20170224/ -skip_if_len
_match
# Set parameters completely through the command line
./hdfs_to_cos_cmd -appid 1252xxxxxx -ak
AKIDVt55xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx -sk
KS08jDVbVElxxxxxxxxxxxxxxxxxxxxxxxxxx -bucket test -cos_path /hdfs
-hdfs_path /data/data -region cn-south -hdfs_conf_file
/home/hadoop/hadoop-2.8.1/etc/hadoop/core-site.xml
```

5. After the command is verified and run, a log will be output as shown below:

```
[Folder Operation Result : [ 53(sum)/ 53(ok) / 0(fail)]]
[File Operation Result: [22(sum)/ 22(ok) / 0(fail) / 0(skip)]]
[Used Time: 3 s]
```

- `sum` indicates the total number of files to be migrated.
- `ok` indicates the number of files successfully migrated.
- `fail` indicates the number of files failed to be migrated.
- `skip` indicates the number of files skipped because they have the same length as the files of the same name in the destination after the `skip_if_len_match` parameter is added.

You can also log in to the COS console to see whether the data has been migrated correctly.

FAQs

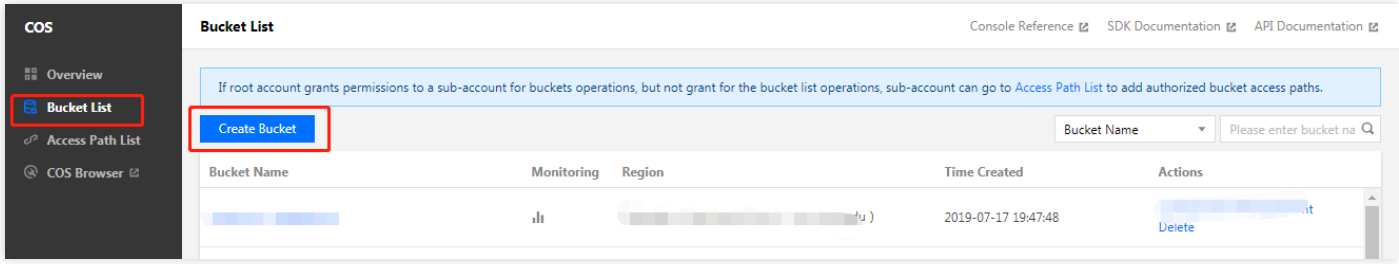
- Make sure that the configuration information is correct, including `appID` , key, bucket, and region. Make sure that the server time is the same as Beijing time (1-minute difference is acceptable. If the difference is too large, reset your server time).
- Make sure that the server for the copy program is accessible to DateNode. The NameNode uses a public IP address and can be accessed, but the DateNode where the obtained block is located uses a private IP address and cannot be accessed; therefore, we recommend you place the copy program in a Hadoop node for execution, so that both the NameNode and DateNode can be accessed.
- In case of a permissions issue, use the current account to download a file with the Hadoop command, check whether everything is correct, and then use the synchronization tool to sync the data in Hadoop.
- Files that already exist in COS are overwritten by default in case of repeated upload, unless you explicitly specify the `-skip_if_len_match` parameter, which indicates to skip files if they have the same length as the existing files.
- The COS path is always considered as a directory, and files that are eventually copied from HDFS will be stored in this directory.

How to View COS Information

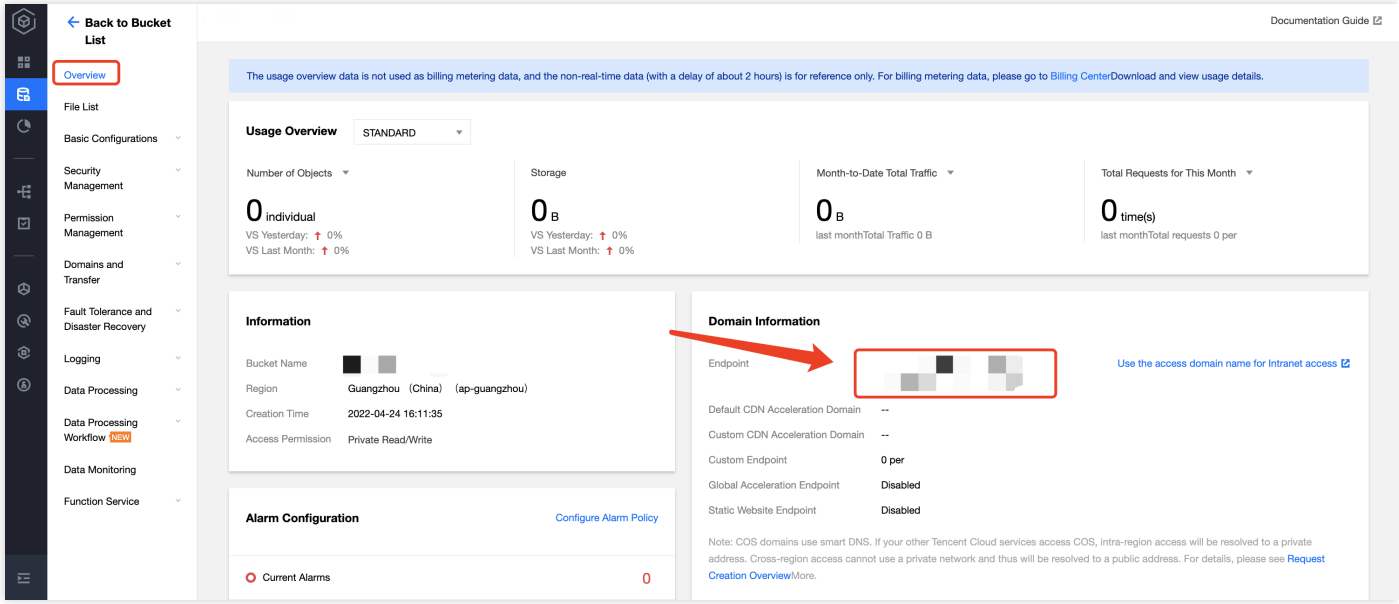
Last updated：2022-05-24 09:31:14

Log in to the COS console and select **Bucket List**.

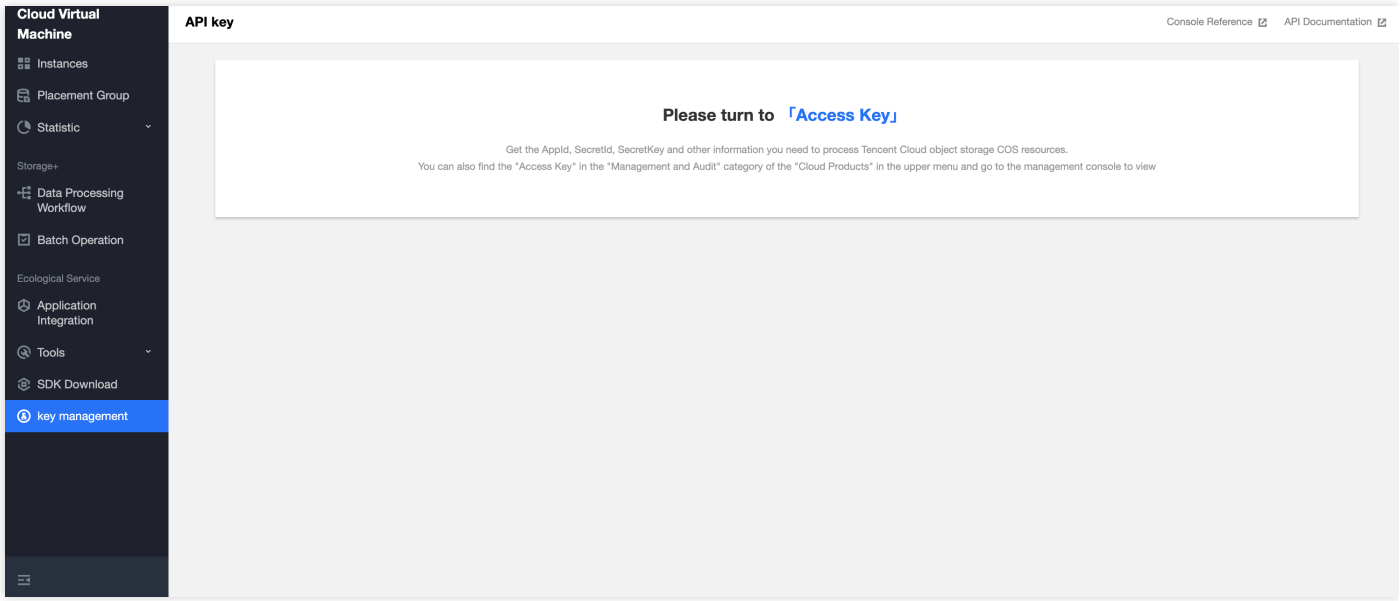
- If you don't have a bucket, you can select **Create Bucket** to create one.



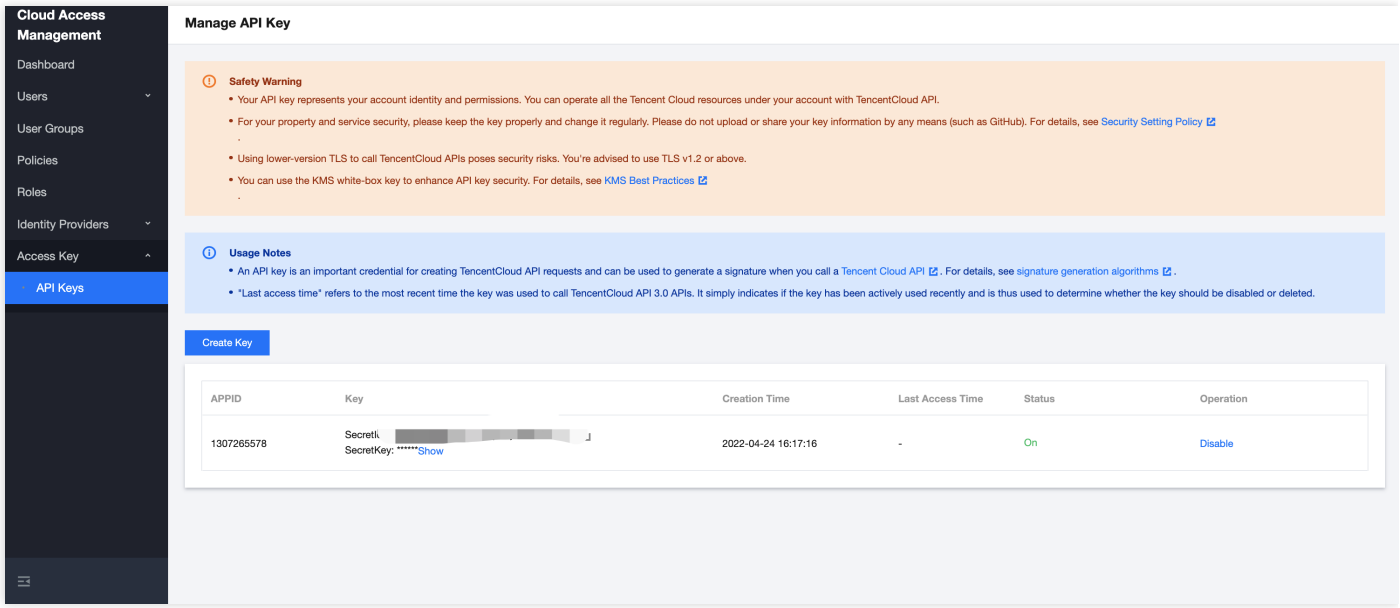
- If you already have a bucket, you can select it and view its access endpoint.



- To query the key, click **API Key** on the left sidebar.



- Click **Cloud API Key**.



Using DistCp to Migrate Data
Basic Description

Last updated：2020-04-03 15:33:49

DistCp (distributed copy) is a tool used for large inter/intra-cluster copying. It uses MapReduce to effect its distribution, error handling and recovery, and reporting. It expands a list of files and directories into input to map tasks, each of which will copy a partition of the files specified in the source list. It is a file migration tool that comes with Hadoop.

Network Interconnection

Last updated：2021-03-19 10:34:56

Migrating Files in a Local Self-built HDFS to EMR

The migration of files in a local self-built HDFS to an EMR cluster requires a direct connection for network connectivity. You can contact Tencent Cloud technical team for assistance.

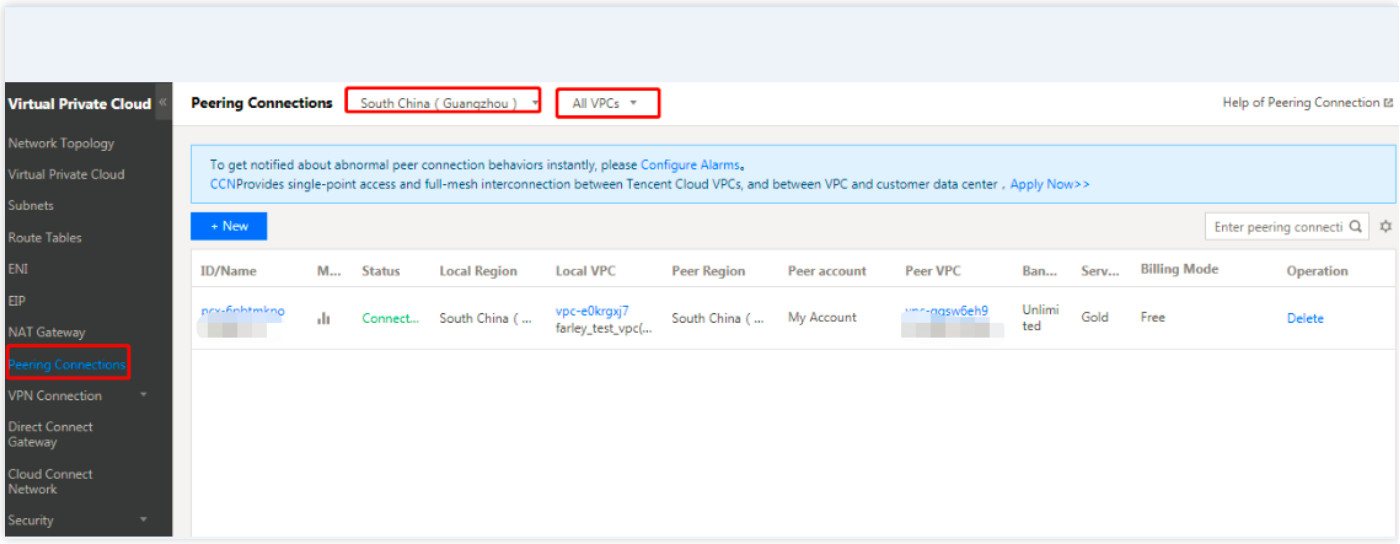
Migrating Files in a Self-built HDFS in CVM to EMR

- If the network where the CVM instance resides and the one where the EMR cluster resides are in the same VPC, the files can be transferred freely.
- Otherwise, a peering connection is required for network connectivity.

Using a peering connection

IP range 1: subnet A 192.168.1.0/24 in VPC1 of Guangzhou.
IP range 2: subnet B 10.0.1.0/24 in VPC2 of Beijing.

1. Log in to the VPC console, go to the **Peering Connections** page, select the region **Guangzhou** at the top of the list, select **VPC1**, and click **+ New**.

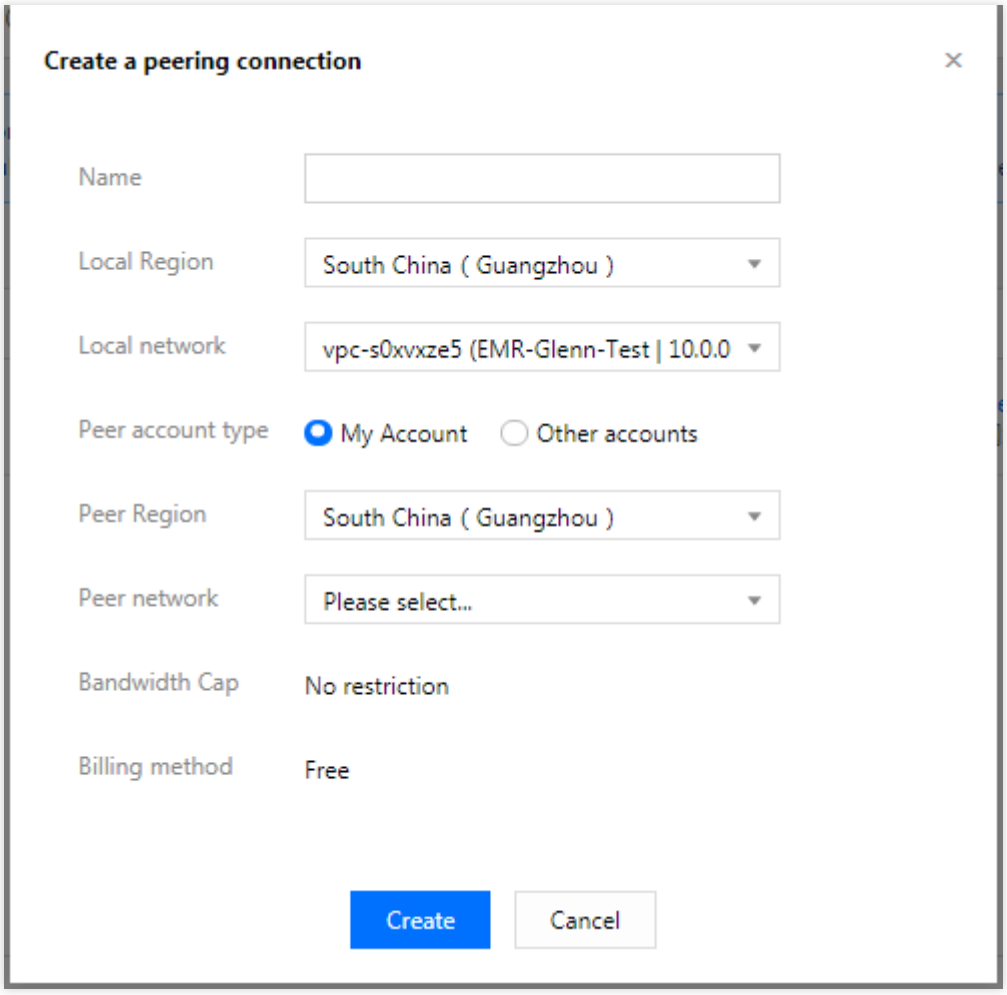


2. On the peering connection creation page, configure the following fields:
   - **Name**: enter the name of the peering connection, e.g., PeerConn.
   - **Local Region**: enter the local region, e.g, Guangzhou.

- **Local Network**: enter the local network, e.g., VPC1.
- **Peer Account Type**: enter the account of the peer network. If the two networks in Guangzhou and Beijing are under the same account, select **My account**; otherwise, select **Other accounts**.

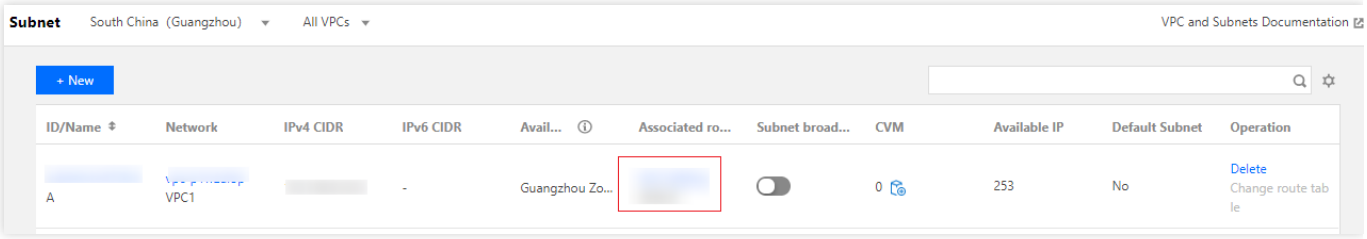
> **Note：**
>
> If both the local network and the peer network are in the same region (e.g., Guangzhou), the communication is free of charge, and there is no need to select the bandwidth cap. Otherwise, fees will be incurred and the bandwidth cap can be set.

- **Peer Region**: enter the peer region, e.g., Beijing.
- **Peer Network**: enter the peer network, e.g., VPC2.

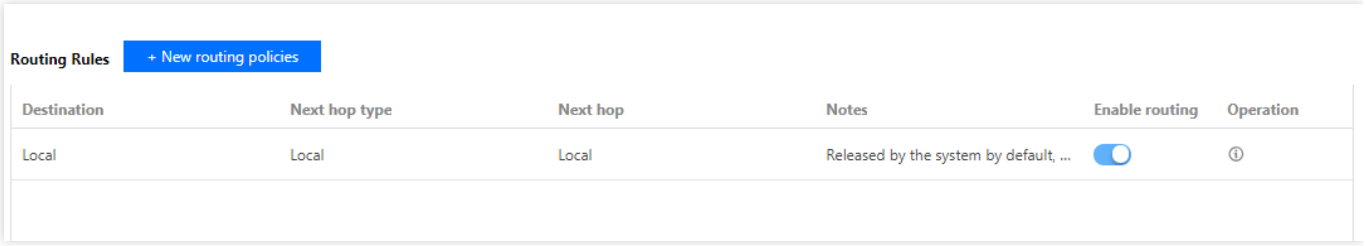


3. A peering connection between VPCs under the same account takes effect immediately after creation. If the VPCs are under different accounts, the peering connection takes effect only after the peer account accepts it. For details, see Creating Intra-account Peering Connection and Creating Cross-account Peering Connection.
4. Configure the local and peer route tables for the peering connection.

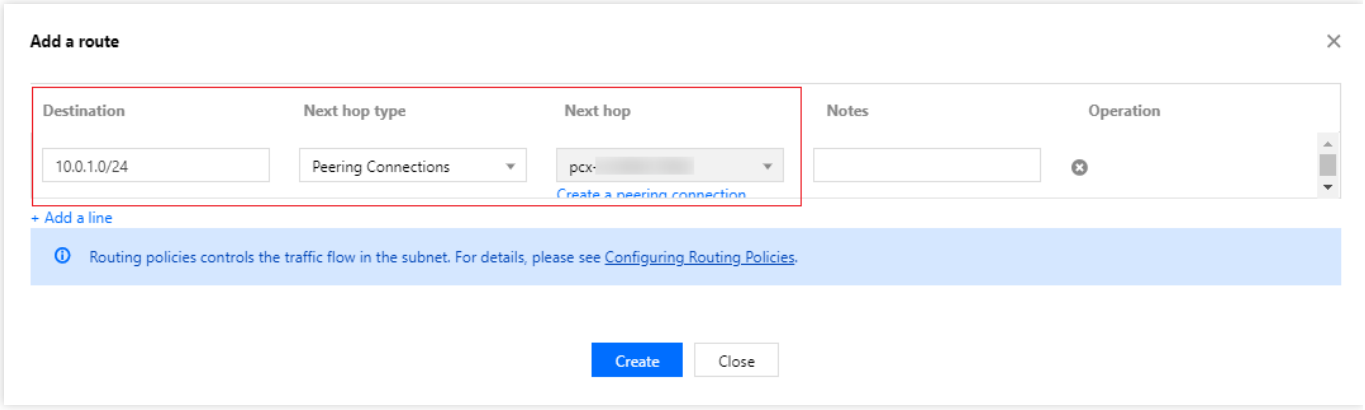
- Log in to the Tencent Cloud console and select **Products** > **Networking** > **Virtual Private Cloud** to go to the VPC console. Click **Subnet** on the left sidebar to go to the management page. Click the ID of the route table associated with the specified subnet (e.g., subnet VPC1 in Guangzhou) on the local end of the peering connection to go to the route table details page.



Click **+ New routing policies**.



- Enter the peer CIDR (for example, the CIDR of VPC2 in Beijing is 10.0.0.1/24) for the destination, select **Peering connections** for the next hop type, and select the created peering connection (PeerConn) for the next hop.



- You've configured the route table from Guangzhou VPC1 to Beijing VPC2 in the previous steps. Now you need to repeat the steps above to configure the route table from Beijing VPC2 to Guangzhou VPC1.
- After the route tables are configured, IP ranges in different VPCs can communicate with each other.

Executing Copy

Last updated：2020-04-03 15:33:50

```
# Copy the specified folder from one cluster to another
hadoop distcp hdfs://nn1:9820/foo/bar hdfs://nn2:9820/bar/foo

# Copy the specified file
hadoop distcp hdfs://nn1:9820/foo/a hdfs://nn1:9820/foo/b hdfs://nn2:9820/bar/foo

# If too many files were specified, use the `-f` parameter to separate them.
```

Precautions

Last updated：2020-04-03 15:33:51

1. For the commands above, the source and destination versions must be the same.
2. You may fail to copy the source file if another client is writing data to it; you will fail to rewrite a file if it is being copied to the destination; you will fail to copy the source file if it was being moved, and you will see an error saying FileNotFoundException.

Hive Migration Guide

Last updated：2023-03-15 10:12:39

Hive migration involves migration of data and metadata. Hive table data is mainly stored in HDFS; therefore, data migration is usually implemented at the HDFS layer. Hive metadata is mainly stored in a relational database and thus can be smoothly migrated to TencentDB for MySQL with guaranteed high availability.

Migrating Hive Metadata

1. Dump the source Hive metastore.

```
mysqldump -hX.X.X.X -uroot -pXXXX --single-transaction --set-gtid-purged=OFF hi
vemetastore > hivemetastore-src.sql
# If GTID is not enabled for MySQL, please delete `--set-gtid-purged=OFF` from
the command
# `X.X.X.X` is the IP address of the database server
# `XXXX` is the database password
# If the database user is not `root`, use the correct username
# `hivemetastore` is the Hive metastore name
```

2. Confirm the default path of the target Hive table in HDFS.
   The default path of the Hive table in HDFS is specified by the `hive.metastore.warehouse.dir` parameter in `hive-site.xml`. If the storage location of the Hive table in HDFS must be the same as that in the source Hive database, you need to modify the configuration file. For example, if the value of `hive.metastore.warehouse.dir` in the source `hive-site.xml` is as follows:

```
<property>
<name>hive.metastore.warehouse.dir</name>
<value>/apps/hive/warehouse</value>
</property>
```

The value of `hive.metastore.warehouse.dir` in the target `hive-site.xml` is as follows:

```
<property>
<name>hive.metastore.warehouse.dir</name>
<value>/usr/hive/warehouse</value>
</property>
```

If the storage location of the target Hive table in HDFS is to be kept the same as that in the source Hive database, change `hive.metastore.warehouse.dir` in the target `hive-site.xml` to the following value:

```
<property>
<name>hive.metastore.warehouse.dir</name>
<value>/apps/hive/warehouse</value>
</property>
```

3. Confirm the `SDS.LOCATION` and `DBS.DB_LOCATION_URI` fields of the target Hive metadata.
   Run the following query statements to get the current `SDS.LOCATION` and `DBS.DB_LOCATION_URI` fields:

```
SELECT DB_LOCATION_URI from DBS;
SELECT LOCATION from SDS;
```

The query result is similar to the following:

```
mysql> SELECT LOCATION from SDS;
+-------------------------------------------------+
| LOCATION |
+-------------------------------------------------+
| hdfs://HDFS2648/usr/hive/warehouse/hitest.db/t1 |
| hdfs://HDFS2648/usr/hive/warehouse/wyp |
+-------------------------------------------------+
mysql> SELECT DB_LOCATION_URI from DBS;
+-----------------------------------------------+
| DB_LOCATION_URI |
+-----------------------------------------------+
| hdfs://HDFS2648/usr/hive/warehouse |
| hdfs://HDFS2648/usr/hive/warehouse/hitest.db |
+-----------------------------------------------+
```

Here, `hdfs://HDFS2648` is the default file system name of HDFS, which is specified by `fs.defaultFS` in `core-site.xml` .

```
<property>
<name>fs.defaultFS</name>
<value>hdfs://HDFS2648</value>
</property>
```

`/usr/hive/warehouse` is the default storage path of the Hive table in HDFS, which is specified by `hive.metastore.warehouse.dir` in `hive-site.xml` . Therefore, you need to modify the `SDS.LOCATION` and `DBS.DB_LOCATION_URI` fields in the SQL file of the Hive metadata and make sure that

the two fields in the imported Hive metastore use the correct path. You can run the following `sed` command to modify SQL files in batches.

```
Replace ip: sed -i 's/oldcluster-ip:4007/newcluster-ip:4007/g' hivemetastore-src.sql
Replace defaultFS: sed -i 's/old-defaultFS/new-defaultFS/g' hivemetastore-src.sql
```

Note：
If some components such as Kudu and HBase are used, and metastore is used as the metadata service, then the `location` field in the target Hive metadata also needs to be changed.

4. Stop the MetaStore, HiveServer2, and WebHcataLog services of the target Hive database.

5. Back up the target Hive metastore.

```
mysqldump -hX.X.X.X -uroot -pXXXX --single-transaction --set-gtid-purged=OFF hivemetastore > hivemetastore-target.sql
# If GTID is not enabled for MySQL, please delete `--set-gtid-purged=OFF` from the command
# `X.X.X.X` is the IP address of the database server
# `XXXX` is the database password
# If the database user is not `root`, use the correct username
# `hivemetastore` is the Hive metastore name
```

6. Drop and create the target Hive metastore.

```
mysql> drop database hivemetastore;
mysql> create database hivemetastore;
```

7. Import the source Hive metastore to the target database.

```
mysql -hX.X.X.X -uroot -pXXXX hivemetastore < hivemetastore-src.sql
# `X.X.X.X` is the IP address of the database server
# `XXXX` is the database password
# If the database user is not `root`, use the correct username
# `hivemetastore` is the Hive metastore name
```

8. Upgrade the Hive metadata.
   If the versions of the target and source Hive databases are the same, you can skip this step; otherwise, query the Hive version in the source and target clusters, respectively.

```
hive --service version
```

The Hive upgrade script is stored in the `/usr/local/service/hive/scripts/metastore/upgrade/mysql/` directory.
Hive does not support upgrade across major versions. For example, if you want to upgrade Hive from 1.2 to 2.3.0, perform the following operations in sequence:

```
upgrade-1.2.0-to-2.0.0.mysql.sql -> upgrade-2.0.0-to-2.1.0.mysql.sql -> upgrade-2
.1.0-to-2.2.0.mysql.sql -> upgrade-2.2.0-to-2.3.0.mysql.sql
```

The main operations in the upgrade script are creating tables, adding fields, and modifying content. If a table or field already exists, the exception for table or field existence can be ignored during the upgrade. For example, you can upgrade Hive from 2.3.3 to 3.1.1 as follows:

```
mysql> source upgrade-2.3.0-to-3.0.0.mysql.sql;
mysql> source upgrade-3.0.0-to-3.1.0.mysql.sql;
```

9. If there is a Phoenix table in the source Hive database, modify the ZooKeeper address of the Phoenix table in the target Hive metadata.
   Run the following query statement to get the `phoenix.zookeeper.quorum` configuration of the Phoenix table.

```
mysql> SELECT PARAM_VALUE from TABLE_PARAMS where PARAM_KEY = 'phoenix.zookeepe
r.quorum';
+-------------------------------------------------+
| PARAM_VALUE |
+-------------------------------------------------+
| 172.17.64.57,172.17.64.78,172.17.64.54 |
+-------------------------------------------------+
```

View the ZooKeeper address of the target cluster, i.e., the value specified in `hbase.zookeeper.quorum` of the `hive-site.xml` configuration file.

```
<property>
<name>hbase.zookeeper.quorum</name>
<value>172.17.64.98:2181,172.17.64.112:2181,172.17.64.223:2181</value>
</property>
```

Modify the ZooKeeper address of the Phoenix table in the target Hive metadata to that of the target cluster.

```
mysql> UPDATE TABLE_PARAMS set PARAM_VALUE = '172.17.64.98,172.17.64.112,172.17.64.223' where PARAM_KEY = 'phoenix.zookeeper.quorum';
```

0. Check the case of table names in the metadata of the target Hive database and change all lowercase table names to uppercase ones. Example:

```
alter table metastore_db_properties rename to METASTORE_DB_PROPERTIES;
```

1. Start the MetaStore, HiveServer2, and WebHcataLog services of the target Hive database.

2. Run simple Hive SQL query statements to check the migration result.

Kudu Data Migration Guide

Last updated：2022-05-16 12:52:26

Kudu can use the rebalancing tool to migrate data.

> Note：
> You can disconnect only one tserver at a time. To disconnect multiple ones, repeat the following steps.

Kudu Migration Based on Rebalancing Tool

1. Make sure that the cluster status is **OK**.

```
/usr/local/service/kudu/bin/kudu cluster ksck 10.0.1.29:7051,10.0.1.16:7051,10.0.1.36:7051
```

```
==================
Warnings:
==================
Some masters have unsafe, experimental, or hidden flags set
Some tablet servers have unsafe, experimental, or hidden flags set

OK
[hadoop@10 bin]$
```

2. Run the ksck command as described in step 1 to get the `uid` of the disconnected nodes.

```
Tablet Server Summary
                UUID                |     Address     | Status  | Location | Tablet Leaders | Active Scanners
------------------------------------+-----------------+---------+----------+----------------+-----------------
 20681b1d6b9942cbab95dded905406ec | 10.0.1.37:7050 | HEALTHY | <none>   |       9        |        0
 6929daf14f8647c89fb8cc51db5d70b6 | 10.0.1.15:7050 | HEALTHY | <none>   |       5        |        0
 b53b28bfad2c41d38d6f08a261ceb486 | 10.0.1.40:7050 | HEALTHY | <none>   |       2        |        0
 be018287364d4443a48ad1bba248c87f | 10.0.1.9:7050  | HEALTHY | <none>   |       0        |        0
 fb9afb1b2989456cac5800bf6990dfea | 10.0.1.45:7050 | HEALTHY | <none>   |       0        |        0
```

Take the `fb9afb1b2989456cac5800bf6990dfea` node as an example.

3. Switch the `fb9afb1b2989456cac5800bf6990dfea` node to the maintenance mode.

```
/usr/local/service/kudu/bin/kudu tserver state enter_maintenance 10.0.1.29:7051
,10.0.1.16:7051,10.0.1.36:7051 fb9afb1b2989456cac5800bf6990dfea
```

4. Run the following rebalancing command.

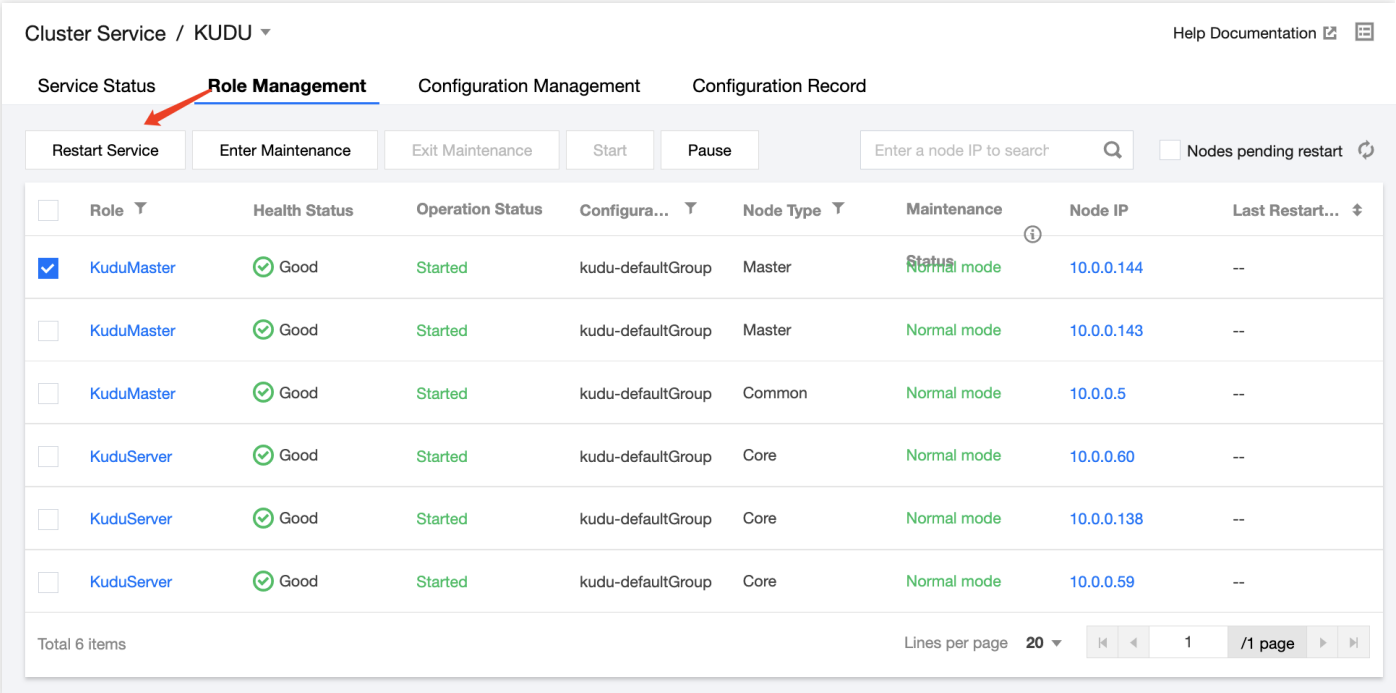
```
/usr/local/service/kudu/bin/kudu cluster rebalance 10.0.1.29:7051,10.0.1.16:705
1,10.0.1.36:7051 --ignored_tservers fb9afb1b2989456cac5800bf6990dfea --move_rep
licas_from_ignored_tservers
```

After the command is executed, run the ksck command again to check whether the status is **OK** before proceeding.

5. Suspend the tserver process at `10.0.1.45` on the `fb9afb1b2989456cac5800bf6990dfea` node. Note that at this point, if you run the ksck command, you will see that the cluster is unhealthy, and you need to restart the tmasters.

```
Tablet Server Summary
              UUID               |     Address     |   Status    | Location | Tablet Leaders | Active Scanners
---------------------------------+-----------------+-------------+----------+----------------+-----------------
 20681b1d6b9942cbab95dded905406ec | 10.0.1.37:7050 | HEALTHY     | <none>   |       9        |       0
 6929daf14f8647c89fb8cc51db5d70b6 | 10.0.1.15:7050 | HEALTHY     | <none>   |       5        |       0
 b53b28bfad2c41d38d6f08a261ceb486 | 10.0.1.40:7050 | HEALTHY     | <none>   |       2        |       0
 be018287364d4443a48ad1bba248c87f | 10.0.1.9:7050  | HEALTHY     | <none>   |       0        |       0
 fb9afb1b2989456cac5800bf6990dfea | 10.0.1.45:7050 | UNAVAILABLE | <none>   | n/a            | n/a
```

6. Restart the masters in the EMR console one by one (rolling restart in the console is not recommended). Then, run the ksck command to check whether the cluster is healthy.



Data Migration
HDFS Data Migration Using COS

Last updated：2023-07-14 10:57:58

If you need to migrate your HDFS raw data to EMR, you can achieve using either of the following: migrate data with Tencent Cloud Object Storage (COS) service as a transfer stop; migrate data with DistCp, a built-in tool of Hadoop for large inter/intra-cluster copying. This document describes how to migrate data with the first method.

Migrating a non-HDFS file

If your source file is not an HDFS file, upload it to COS via the COS console or API, and then analyze it in the EMR cluster.

Migrating an HDFS file

1. Get the COS migration tool.
   Get the migration tool hdfs_to_cos_tools. For more migration tools, see Tool Overview.

2. Configure the tool.
   All configuration files are stored in the `conf` directory of the tool directory. Copy the `core-site.xml` file of the HDFS cluster to be synced to `conf`, which contains the configuration information of the NameNode. Edit the configuration file `cos_info.conf` by including your `appid`, bucket, region, and key information.

   Note：
   - We recommend you use a sub-account key and follow the principle of least privilege to avoid leaking resources besides your buckets and objects.
   - If you need to use a permanent key, we recommend you follow the principle of least privilege to limit the scope of permission for the permanent key.

   Command parameter descriptions:

```
-ak <ak> the cos secret id // Your `SecretId`. We recommend you use a sub-account key and follow the principle of least privilege to reduce risks. For information on how to get a sub-account key, visit https://cloud.tencent.com/document/product/598/37140.
-appid,--appid <appid> the cos appid
-bucket,--bucket <bucket_name> the cos bucket name
```

```
-cos_info_file,--cos_info_file <arg> the cos user info config default is ./conf
/cos_info.conf
-cos_path,--cos_path <cos_path> the absolute cos folder path
-h,--help print help message
-hdfs_conf_file,--hdfs_conf_file <arg> the hdfs info config default is ./conf/c
ore-site.xml
-hdfs_path,--hdfs_path <hdfs_path> the hdfs path
-region,--region <region> the cos region. legal value cn-south, cn-east, cn-nor
th, sg
-sk <sk> the cos secret key // Your `SecretKey`. We recommend you use a sub-acc
ount key and follow the principle of least privilege to reduce risks. For infor
mation on how to get a sub-account key, visit https://cloud.tencent.com/documen
t/product/598/37140.
-skip_if_len_match,--skip_if_len_match skip upload if hadoop file length match
cos
```

3. Execute data migration.

```
# All operations must be performed in the tool directory. If both configuration
files and command line parameters are set, the latter will prevail
./hdfs_to_cos_cmd -h
# Copy from HDFS to COS (if a file already exists in COS, it will be overwritte
n)
./hdfs_to_cos_cmd --hdfs_path=/tmp/hive --cos_path=/hdfs/20170224/
# Copy from HDFS to COS, and if a file to be copied is of the same length as a
file in COS, then it is skipped (this is suitable for repeated copy)
# Only the length is checked here, as the overheads would be very high if the d
igests of files in Hadoop are to be calculated
./hdfs_to_cos_cmd --hdfs_path=/tmp/hive --cos_path=/hdfs/20170224/ -skip_if_len
_match
# Set parameters completely through the command line
./hdfs_to_cos_cmd -appid 1252xxxxxx -ak
AKIDVt55xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx -sk
KS08jDVbVElxxxxxxxxxxxxxxxxxxxxxxxxxxxx -bucket test -cos_path /hdfs
-hdfs_path /data/data -region cn-south -hdfs_conf_file
/home/hadoop/hadoop-2.8.1/etc/hadoop/core-site.xml
```

4. After the command is verified and run, a log will be generated as shown below:

```
[Folder Operation Result : [ 53(sum)/ 53(ok) / 0(fail)]]
[File Operation Result: [22(sum)/ 22(ok) / 0(fail) / 0(skip)]]
[Used Time: 3 s]
```

- `sum` indicates the total number of files to be migrated.
- `ok` indicates the number of files successfully migrated.
- `fail` indicates the number of files failed to be migrated.
- `skip` indicates the number of files skipped because they have the same length as the files of the same name in the destination after the `skip_if_len_match` parameter is added.

You can also log in to the COS console to check whether the data has been migrated correctly. For how to use COS, see Console.

FAQ

- Make sure that the configuration information is correct, including `appID` , key, bucket, and region. Make sure that the server time is the same as Beijing time (1-minute difference is acceptable. If the difference is too large, reset your server time).
- Make sure that the server for the copy program is accessible to DateNode. The NameNode uses a public IP address and can be accessed, but the DateNode where the obtained block is located uses a private IP address and cannot be accessed; therefore, we recommend you place the copy program in a Hadoop node for execution, so that both the NameNode and DateNode can be accessed.
- In case of a permissions issue, use the current account to download a file with the Hadoop command, check whether everything is correct, and then use the synchronization tool to sync the data in Hadoop.
- Files that already exist in COS are overwritten by default in case of repeated upload, unless you explicitly specify the `-skip_if_len_match` parameter, which indicates to skip files if they have the same length as the existing files.
- The COS path is always considered as a directory, and files that are eventually copied from HDFS will be stored in this directory.

HDFS Data Migration Using DistCp

Last updated：2023-07-14 10:57:58

If you need to migrate your HDFS raw data to EMR, you can achieve this using either of the following: migrate data with Tencent Cloud Object Storage (COS) service as a transfer stop; migrate data with DistCp, a built-in tool of Hadoop for large inter/intra-cluster copying. This document describes how to migrate data with the second method.

DistCp (distributed copy) is a file migration tool that comes with Hadoop. It uses MapReduce to effect its distribution, error handling and recovery, and reporting. It expands a list of files and directories into input to map tasks, each of which will copy a partition of the files specified in the source list. To use DistCp, your cluster and the EMR cluster must be connected over network.
To migrate data with DistCp, perform the following steps:

Step 1. Configure a Network

Migrating local self-built HDFS files to EMR

The migration of local self-built HDFS files to an EMR cluster requires a direct connection for network connectivity. You can contact Tencent Cloud technical team for assistance.

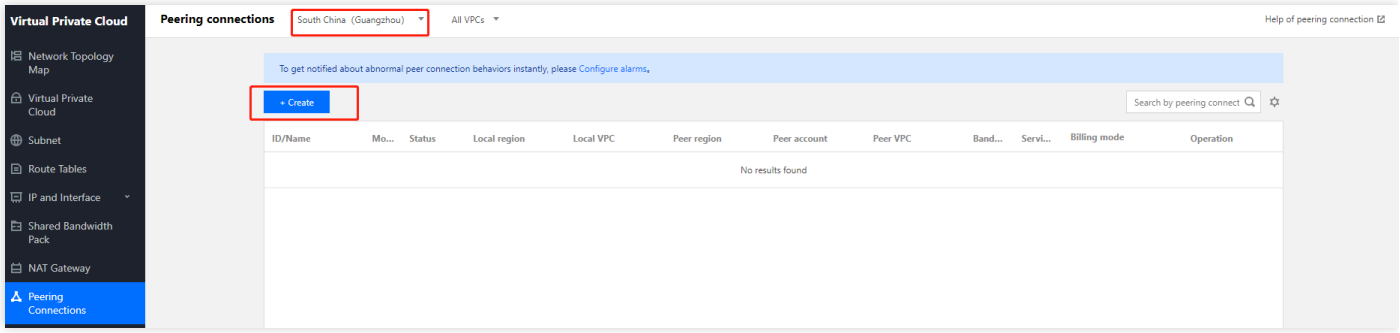
Migrating self-built HDFS files in CVM to EMR

- If the network where the CVM instance resides and the one where the EMR cluster resides are in the same VPC, the files can be transferred freely.
- Otherwise, a peering connection is required for network connectivity.

Using a peering connection

IP CIDR block 1: Subnet A 192.168.1.0/24 in VPC1 of Guangzhou.
IP CIDR block 2: Subnet B 10.0.1.0/24 in VPC2 of Beijing.

1. Log in to the VPC console, enter the **Peering Connections** page, select the region **Guangzhou** at the top of the page, select **VPC1**, and click **+ Create**.



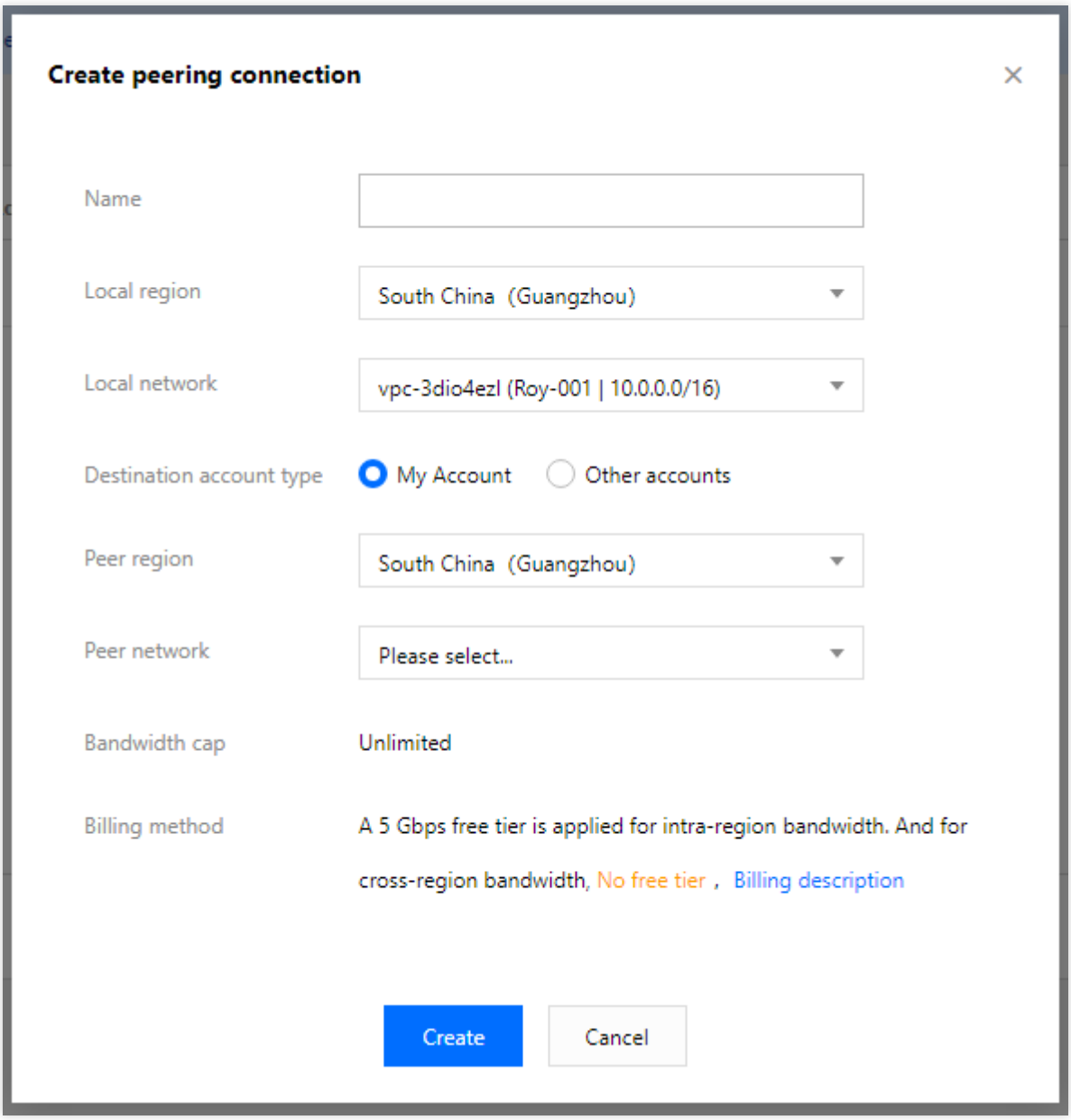
2. On the peering connection creation page, configure the following fields:

- **Name**: Enter a peering connection name, such as PeerConn.
- **Local region**: Enter a local region, such as Guangzhou.
- **Local network**: Enter a local network, such as VPC1.
- **Destination account type**: Select the account of the peer network. If the two networks in Guangzhou and Beijing are under the same account, select **My account**; otherwise, select **Other accounts**.

> Note：
> If both the local and peer networks are in the same region (such as Guangzhou), the communication is free of charge, and you do not need to set the bandwidth cap. Otherwise, fees will be incurred and you can set the bandwidth cap.

- **Peer region**: Enter a peer region, such as Beijing.

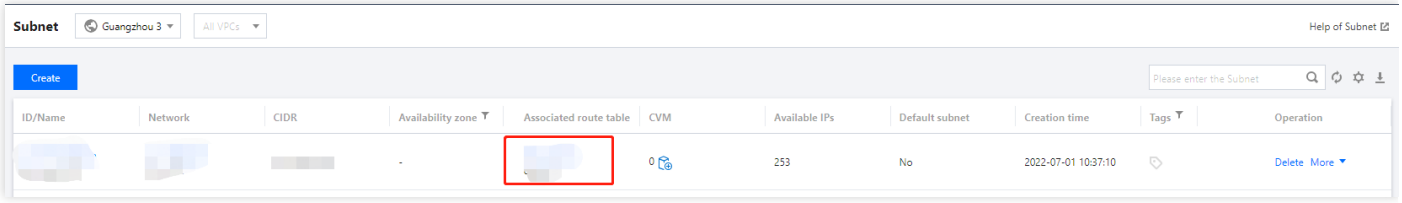
- **Peer network**: Enter a peer network, such as VPC2.



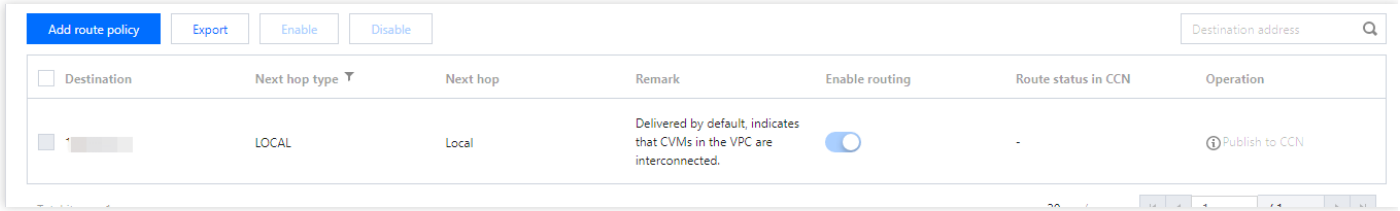
3. A peering connection between VPCs under the same account takes effect immediately after creation. If the VPCs are under different accounts, the peering connection takes effect only after the peer account accepts it. For details, see Creating Intra-account Peering Connection and Creating Cross-account Peering Connection.
4. Configure the local and peer route tables for the peering connection.

- Log in to the VPC console and select **Subnet** to enter the subnet management page. Click the ID of the route table associated with the specified subnet (such as subnet VPC1 in Guangzhou) on the local end of the peering

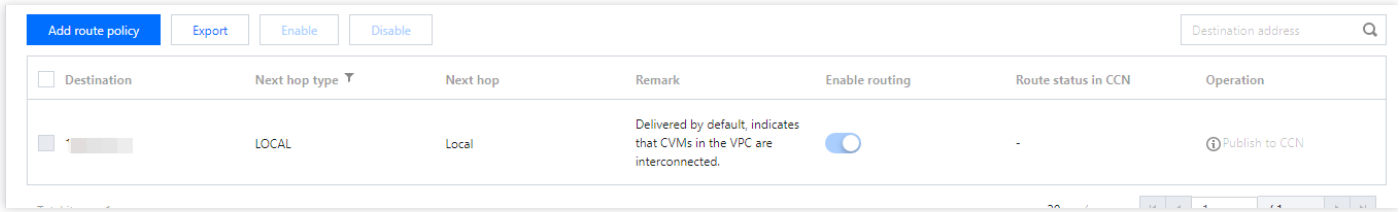
connection to enter the route table details page.



- Click **Add route policy**.



- Enter the destination CIDR block (such as 10.0.1.0/24 for VPC2 in Beijing), select **Peering connections** for the next hop type, and select the created peering connection (PeerConn) for the next hop.



- You've configured the route table from Guangzhou VPC1 to Beijing VPC2 in the previous steps. Now you need to repeat the steps above to configure the route table from Beijing VPC2 to Guangzhou VPC1.
- After the route tables are configured, IP CIDR blocks in different VPCs can communicate with each other.

Step 2. Execute copying

```
# Copy the specified folder from one cluster to another
hadoop distcp hdfs://nn1:9820/foo/bar hdfs://nn2:9820/bar/foo

# Copy the specified file
hadoop distcp hdfs://nn1:9820/foo/a hdfs://nn1:9820/foo/b hdfs://nn2:9820/bar/foo

# If too many files need to be specified, use -f parameter to separate them.
```

Note：

- For the commands above, the source and destination versions must be the same.
- The copying will fail if another client is writing data to the source file or the source file was moved (the `FileNotFoundException` error message will occur); rewriting the source file will fail if it is being copied to the destination.