

Koo Command Line Interface

User Guide

Issue	01
Date	2026-09-02



Copyright © Huawei Cloud Computing Technologies Co., Ltd. 2026. All rights reserved.

No part of this document may be reproduced or transmitted in any form or by any means without prior written consent of Huawei Cloud Computing Technologies Co., Ltd.

Trademarks and Permissions



HUAWEI and other Huawei trademarks are the property 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 Cloud 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.

Huawei Cloud Computing Technologies Co., Ltd.

Address: Huawei Cloud Data Center Jiaoxinggong Road
Qianzhong Avenue
Gui'an New District
Gui Zhou 550029
People's Republic of China

Website: <https://www.huaweicloud.com/intl/en-us/>

Contents

1 Configuration Management.....	1
1.1 Configuration Commands.....	1
1.2 Initializing Configurations.....	1
1.3 Adding or Modifying a Profile.....	2
1.3.1 Profile Overview.....	2
1.3.2 Setting System Parameters in a Profile.....	5
1.3.3 Setting Custom Parameters in a Profile.....	7
1.4 Logging In via SSO.....	8
1.5 Querying the Profile List.....	9
1.5.1 Querying System Parameters in the Profile List.....	9
1.5.2 Querying Custom Parameters in the Profile List.....	11
1.6 Querying a Profile.....	12
1.6.1 Querying System Parameters in a Profile.....	12
1.6.2 Querying Custom Parameters in a Profile.....	12
1.7 Deleting a Profile.....	12
1.7.1 Deleting a Profile and Its Custom Parameters.....	13
1.7.2 Deleting Custom Parameters in a Profile.....	13
1.8 Checking the Configuration File Format.....	13
1.9 Deleting All Profiles.....	13
2 HTTP Proxy.....	14
2.1 Configuring the HTTP Proxy.....	14
2.2 Resolving the x509 Certificate Error.....	17
3 Commonly Used Commands.....	18
3.1 Viewing Help Information.....	18
3.2 Turning On/Off Autocomplete.....	18
3.3 Using Interactive Prompts.....	19
3.4 Managing Metadata.....	21
3.5 Querying the Current Version.....	22
3.6 Upgrading the Version.....	22
3.7 Managing Logs.....	23
3.8 Managing Data in OBS.....	23
3.9 Managing CCE Clusters.....	39

4 Options.....	40
4.1 Option Overview.....	40
4.2 Printing Help Information.....	44
4.3 Printing Debugging Information.....	44
4.4 Generating API Input Parameter Skeleton in JSON Format.....	45
4.5 Checking Command.....	47
4.6 Specifying Region.....	48
4.7 Calling APIs with AK/SK in Non-configuration Mode.....	48
4.8 Calling APIs Using an AssumeRole in Non-configuration Mode.....	49
4.9 Specifying Account ID.....	49
4.10 Specifying Profile.....	50
4.11 Specifying Authentication Mode.....	50
4.12 Specifying Output Format.....	50
4.13 Passing API Parameters with JSON File.....	54
4.14 Specifying Request Timeouts.....	56
4.15 Specifying Retry Count.....	57
4.16 Skipping HTTPS Request Certificate Verification.....	57
4.17 Specifying a Domain Name.....	58
4.18 Polling Results.....	58
4.19 Specify a Signing Algorithm.....	59
4.20 Specifying a Project ID.....	59
5 Obtaining CLI Examples on API Explorer.....	60
6 Using KooCLI in Non-configuration Mode.....	61
6.1 Introduction.....	61
6.2 AK/SK Authentication.....	61
6.3 ecsAgency Authentication.....	62
7 Obtaining Authentication Information.....	63
7.1 Obtaining a Permanent AK/SK.....	63
7.2 Obtaining an Account ID and Project ID.....	64
7.3 Obtaining a Region.....	65
7.4 Obtaining a Temporary AK/SK and SecurityToken.....	65

1 Configuration Management

1.1 Configuration Commands

Koo Command Line Interface (KooCLI) configuration commands are used to initialize configurations, and add, delete, modify, and query profiles. The following table lists the commands.

System Command	Function
hcloud configure init	Initialize configurations.
hcloud configure set	Add or modify a profile.
hcloud configure sso	Log in through single sign-on (SSO).
hcloud configure list	Query the profile list.
hcloud configure show	Query a specific profile.
hcloud configure delete	Delete a specific profile.
hcloud configure test	Check the configuration file format.
hcloud configure clear	Delete all profiles.

1.2 Initializing Configurations

The KooCLI initialization command is used to store the **permanent** AK/SK and region information you often use in the configuration file, eliminating the need for frequently entering the information during operations. Run the following command to initialize configurations:

hcloud configure init

After you type in the preceding command and press **Enter**, the system prompts you to enter the **access key ID** (required), **secret access key** (required), and default or common **region** (optional).

```
hcloud configure init
Starting initialization. 'Secret Access Key' is anonymized. To obtain the parameter, see 'https://
support.huaweicloud.com/intl/en-us/usermanual-hcli/hcli_09.html'.
Access Key ID [required]: *****
Secret Access Key [required]: ****
Secret Access Key (again): ****
Region: ap-southeast-1

*****
****                               ****
****      Initialization successful      ****
****                               ****
*****
```

1.3 Adding or Modifying a Profile

1.3.1 Profile Overview

To manage resources in multiple regions or projects or use multiple accounts, add profiles to store fixed information in different scenarios. If you have configured multiple profiles, use **--cli-profile** to specify the name of the profile to be configured or used.

The following table lists the KooCLI profile parameters that can be modified.

Table 1-1 Profile parameters

Parameter	Description
cli-profile	Name of a profile. When you modify a profile without specifying this parameter, the default profile will be modified. To view the content of the default profile, run the hcloud configure show command.
cli-mode	Authentication mode. Options: • AKSK: permanent AK/SK (access key) or temporary AK/SK and SecurityToken (security credentials). • ecsAgency: Temporary authentication information is automatically obtained when you use KooCLI to call cloud service APIs on an Elastic Cloud Server (ECS). • SSO: After you enable IAM Identity Center and complete configurations, your users can log in to the unified portal with their username and password and access allocated resources under multiple accounts. They do not need to log in again before the authentication information expires. • AssumeRole: After a trust relationship is established between two accounts, the delegated party can be authenticated using the agency to manage cloud services and resources of the delegating party.
cli-region	Region.
cli-access-key	Access key ID (AK). It must be modified together with SK.

Parameter	Description
cli-secret-key	Secret access key (SK). It must be modified together with AK.
cli-security-token	Temporary security credential. It is required when you use a temporary AK/SK for identity authentication. It must be reconfigured when it expires. Otherwise, the authentication cannot be completed when you call cloud service APIs.
cli-lang	Language of KooCLI. Options: • cn: Chinese • en: English
cli-offline	Specifies whether to use KooCLI offline mode. Options: • true (offline mode) • false (default value; online mode)
cli-project-id	Project ID.
cli-domain-id	ID of the account to which an IAM user belongs. This parameter is required when you call APIs of global services in AK/SK mode.
cli-custom	Specifies whether to configure custom parameters. Options: • true: Configure the custom parameters in the profile. • false: Default value, which means to configure the system parameters in the profile.
cli-read-timeout	I/O timeout, in seconds. The default value is 10 , and the minimum value is 1 .
cli-connect-timeout	Request connection timeout, in seconds. The default value is 5 , and the minimum value is 1 .
cli-retry-count	Number of connection attempts. The value ranges from 0 to 5, and the default value is 0 .
cli-skip-secure-verify	Specifies whether to skip HTTPS certificate verification (not recommended). Options: • true: HTTPS certificate verification is skipped. This is not recommended for security purposes. • false: Default value, which means that HTTPS certificate verification is required.
cli-agency-domain-name	Account name of a delegating party. This parameter must be used together with cli-agency-name .
cli-agency-domain-id	Account ID of the delegating party. This parameter must be used together with cli-agency-name .
cli-agency-name	Agency name. It must be used together with cli-agency-domain-id or cli-agency-domain-name .

Parameter	Description
cli-source-profile	The profile that stores the authentication information of the delegated party. The value of cli-source-profile cannot be the current profile.
cli-agree-privacy-statement	Whether to agree to the privacy statement. When using KooCLI for the first time, you will be prompted to read and agree to the privacy statement. In some special scenarios such as command execution with automation scripts, use this parameter to agree to the privacy statement. Options: • true: Agree with the privacy statement. • false (default): Disagree with the privacy statement.
cli-warning	Whether to display warnings during command execution. Disabling warnings can avoid interfering with result resolution during command execution with automation scripts. Options: • true (default): Display warnings during command execution. • false: Display no warnings during command execution.
cli-auth-encrypt	Whether to encrypt authentication information in the configuration file before storage. Options: • true (default): The authentication information is encrypted before storage. • false: The authentication information is not encrypted before storage.
cli-sso-start-url	User portal URL, which is generated after you enable IAM Identity Center.
cli-sso-region	Region where your IAM Identity Center instance is enabled.
cli-sso-account-name	Account name.
cli-sso-permission-set-name	Permission set name.

 NOTE

- The value of **--cli-custom** indicates the type of the parameter to be set in the profile. If **--cli-custom** is set to **false**, the system parameters will be set. If **--cli-custom** is set to **true**, the custom parameters will be set.
- When you set a system parameter for a profile that does not exist, the profile is created; if you do so for an existing profile, the profile is modified.
- You cannot set custom parameters for a profile that does not exist. You cannot set both system parameters and custom parameters in the same command.
- After you set system parameters or custom parameters for a profile, the profile then becomes **the default**.

Command for adding or modifying a profile:

```
hcloud configure set --param1=paramValue1 --param2=paramValue2 ...
```

1.3.2 Setting System Parameters in a Profile

The following are examples of setting system parameters in a profile.

- Adding a profile

When adding a profile, use **--cli-profile** to declare the profile name. For example:

- Add a profile with AK/SK authentication. Authentication parameters: permanent AK (**cli-access-key**) and SK (**cli-secret-key**).

```
hcloud configure set --cli-profile=testAKSK --cli-mode=AKSK --cli-region=ap-southeast-1 --cli-access-key=***** --cli-secret-key=***** --cli-project-id=068119468*****af89d2e --cli-domain-id=094518e46*****cfbc4c0 --cli-read-timeout=10 --cli-connect-timeout=5
```

- Add a profile with AK/SK authentication. Authentication parameters: temporary AK (**cli-access-key**), SK (**cli-secret-key**), and SecurityToken (**cli-security-token**).

```
hcloud configure set --cli-profile=testAKSKST --cli-mode=AKSK --cli-region=ap-southeast-1 --cli-access-key=***** --cli-secret-key=***** --cli-security-token=***** --cli-project-id=068119468*****af89d2e --cli-domain-id=094518e46*****cfbc4c0 --cli-read-timeout=10 --cli-connect-timeout=5
```

For a profile with AK/SK authentication mode, if the **cli-security-token** parameter is configured in the profile, **cli-access-key** and **cli-secret-key** are temporary AK/SK. Otherwise, they are permanent AK/SK.

- Add a profile with ecsAgency authentication. No authentication parameters are required.

```
hcloud configure set --cli-profile=testEcsAgency --cli-mode=ecsAgency --cli-region=ap-southeast-1 --cli-read-timeout=10 --cli-connect-timeout=5
```

When you use KooCLI to call cloud service APIs on an ECS, the CLI automatically obtains temporary authentication information based on the ECS agency.

- Add a profile with AssumeRole authentication. Authentication parameters: delegating account name (**cli-agency-domain-name**) or ID (**cli-agency-domain-id**), agency name (**cli-agency-name**), and the profile (**cli-source-profile**) that stores the authentication information of the delegated account.

- Create a profile or use an existing one to store the authentication information of the delegated account. The profile uses AK/SK authentication.

```
hcloud configure set --cli-profile=testAKSK --cli-mode=AKSK --cli-region=ap-southeast-1 --cli-access-key=***** --cli-secret-key=*****
```

- Create another profile that uses AssumeRole authentication. Use the profile configured in the previous step as the value of **cli-source-profile**.

```
hcloud configure set --cli-profile=testAssumeRole --cli-mode=AssumeRole --cli-region=ap-southeast-1 --cli-agency-domain-id=13534326*****5cf67b --cli-agency-name=***** --cli-source-profile=testAKSK --cli-read-timeout=10 --cli-connect-timeout=5
```

When you add a profile with AssumeRole authentication, both **cli-agency-domain-id** (or **cli-agency-domain-name**) and **cli-agency-name** must be configured. The value of **cli-profile** cannot be the same as that of **cli-source-profile** in the command. The profile specified by **cli-source-profile** must exist.

- Add a profile with SSO authentication. Authentication parameters: user portal URL (**cli-sso-start-url**), region of an IAM Identity Center instance (**cli-sso-region**), account name (**cli-sso-account-name**), and permission set name (**cli-sso-permission-set-name**). **cli-sso-account-name** and **cli-sso-permission-set-name** are optional.
 - i. Create a profile that uses SSO authentication.

```
hcloud configure set --cli-profile=testSSO --cli-mode=SSO --cli-region=ap-southeast-1 --cli-sso-start-url=https://idcenter.huaweicloud.com/d-3*****6/portal --cli-sso-region=ap-southeast-1 --cli-sso-account-name=ACCOUNT_01 --cli-sso-permission-set-name=PERMISSION_01
```
 - ii. Log in with SSO. You will be prompted to enter an account name and permission set name if **cli-sso-account-name** and **cli-sso-permission-set-name** are not set in step 1.

```
hcloud configure sso --cli-profile=testSSO
```
- Changing the **default profile**

If you have configured multiple profiles, you can run the following command to change the default profile:

```
hcloud configure set --cli-profile=test
```
- Changing the language

This change will take effect for all profiles. Run the following command to change the language:

```
hcloud configure set --cli-lang=cn
```
- Switching between online and offline modes

This change will take effect for all profiles. Run the following command to switch to the offline mode:

```
hcloud configure set --cli-offline=true
```
- Configuring whether to display warnings

This change will take effect for all profiles. Disabling warnings can avoid interfering with result resolution during command execution with automation scripts. To disable warnings, run the following command:

```
hcloud configure set --cli-warning=false
```
- Agreeing to the privacy statement

This change will take effect for all profiles. When using KooCLI for the first time, you will be prompted to read and agree to the privacy statement. In some special scenarios such as command execution with automation scripts, run the following command to agree to the privacy statement:

```
hcloud configure set --cli-agree-privacy-statement=true
```
- Configuring whether to encrypt authentication information in the configuration file before storage

This change will take effect for all profiles. Run the following command to skip encrypting authentication information in the configuration file before storage:

```
hcloud configure set --cli-auth-encrypt=false
```
- Changing the authentication mode of a profile

A profile can contain parameters of multiple authentication modes. Run the following command to specify which authentication mode in a profile is to be used:

```
hcloud configure set --cli-profile=test --cli-mode=AKSK
```
- Changing the region of a profile

```
hcloud configure set --cli-profile=test --cli-region=ap-southeast-1
```

- Modifying the **AK/SK** of a profile

AK and SK must be modified at the same time.

```
hcloud configure set --cli-profile=test --cli-access-key=***** --cli-secret-key=*****
```

- Changing the project ID of a profile

A project ID must be specified when you call the APIs of certain cloud services. To call these APIs, **obtain project IDs** and add those you will often use to a profile.

```
hcloud configure set --cli-profile=test --cli-project-id=068119468*****af89d2e
```

- Modifying the request connection timeout of a profile

```
hcloud configure set --cli-profile=test --cli-connect-timeout=5
```

- Modifying the I/O timeout of a profile

```
hcloud configure set --cli-profile=test --cli-read-timeout=10
```

- Modifying the maximum number of connection attempts of a profile

```
hcloud configure set --cli-profile=test --cli-retry-count=3
```

 CAUTION

Setting the retry count with **--cli-retry-count** may cause idempotence and cyclic API calls. Exercise caution when using this option for resource creation APIs.

-
- Changing the configuration of skipping HTTPS certificate verification in a profile

```
hcloud configure set --cli-profile=test --cli-skip-secure-verify=true
```

 NOTE

You can modify multiple parameters of a profile at a time. For example, to modify **cli-project-id** and **cli-region**, run the following command:

```
hcloud configure set --cli-profile=test --cli-project-id=0681194*****f89d2e --cli-region=ap-southeast-1
```

1.3.3 Setting Custom Parameters in a Profile

KooCLI allows you to set custom parameters in a profile. This helps you manage parameters that are frequently used in commands, have long values, or have values to be encrypted. Run the following command to enter the interactive mode and add or modify custom parameters in a profile:

hcloud configure set [--cli-profile={profileName}] --cli-custom=true

```
hcloud configure set --cli-profile=test --cli-custom=true
1. Enter the custom parameter name: projectId
2. Encrypt the custom parameter for storage? (y/N): n
3. Enter the parameter value: 068100000000000000000000f89d2e
Custom parameter saved. To stop setting more custom parameters, press `Ctrl+C`.
1. Enter the custom parameter name: password
2. Encrypt the custom parameter for storage? (y/N): y
3. Enter the parameter value: ****
Custom parameter saved. To stop setting more custom parameters, press `Ctrl+C`.
```

A custom parameter cannot exceed 128 characters. During interactive setting, determine whether to encrypt the custom parameter for storage according to your requirements. To ensure the security of your custom parameters that need to be

encrypted for storage, the system **anonymizes** the parameter values you enter. The entered characters are displayed as asterisks (****) after you press **Enter**.

To use a custom parameter, replace the original value in the command with **custom.\${Name of the custom parameter}**. For example:

```
hcloud ECS NovaListServers --cli-profile=test --project_id="custom.projectId"
```

When you use an encrypted custom parameter, KooCLI **anonymizes** the parameter value in places where the value may be displayed, for example, in the printed request URL of debugging information or in printed body parameters in dryrun mode.

1.4 Logging In via SSO

The SSO login command of KooCLI stores users' authentication information in the configuration file, eliminating the need for frequently entering the information during operations. Run the following command to implement SSO login:

hcloud configure sso

After pressing **Enter**, enter an SSO profile name (name of the profile to save after SSO login, required), SSO start URL (portal URL, required), SSO region (region where the **IAM Identity Center** instance is enabled, required), and **Region** (default or common cloud service region, optional) as prompted.

```
hcloud configure sso
? Input SSO profile name [required]: sso
? Input SSO start URL [required]: https://idcenter.huaweicloud.com/d-3*****6/portal
? Input SSO region [required]: ap-southeast-1
? Input Region: ap-southeast-1
Browser opened. Log in via SSO...
? Choose account name: ACCOUNT_01
? Choose permission set name: PERMISSION_01
SSO login successful.
```

NOTICE

- The SSO login page cannot be automatically displayed in environments without a browser, such as Linux. In this case, open the printed URL with a browser in another environment.
- After web page login, use the up and down arrow buttons to select an account name and permission set name, or enter a keyword, and press **Enter**.
- After SSO login, you can only call APIs of the **specified services**.

After the login is complete, run the following command to view the saved authentication information and its expiration time:

hcloud configure show --cli-profile=\${profileName}

```
hcloud configure show --cli-profile=sso
{
  "name": "sso",
  "mode": "SSO",
  "accessKeyId": "",
  "secretAccessKey": "",
  "securityToken": "",
}
```

```
"region": "ap-southeast-1",
"projectId": "",
"domainId": "",
"skipSecureVerify": "false",
"readTimeout": 10,
"connectTimeout": 5,
"retryCount": 0,
"ssoParam": {
  "startUrl": "https://idcenter.huaweicloud.com/d-3*****6/portal",
  "ssoRegion": "ap-southeast-1"
},
"ssoAuth": {
  "accessTokenResult": {
    "accessToken": "****",
    "expiresAt": "2024-05-08 00:52:50"
  },
  "accountId": "0cb8*****c3a0",
  "agencyUrn": "iam::0cb8*****c3a0:agency:Sys*****_*****_dc*****e8"
}
}
```

SSO logout is supported. Run the command below to clear the authentication information in the specified profile. To re-authenticate, run **hcloud configure sso --cli-profile=\${profileName}**.

hcloud configure sso --cli-profile=\${profileName} --cli-operate=logout

```
hcloud configure show --cli-profile=sso
{
  "name": "sso",
  "mode": "SSO",
  "accessKeyId": "",
  "secretAccessKey": "",
  "securityToken": "",
  "region": "ap-southeast-1",
  "projectId": "",
  "domainId": "",
  "skipSecureVerify": "false",
  "readTimeout": 10,
  "connectTimeout": 5,
  "retryCount": 0,
  "ssoParam": {
    "startUrl": "https://idcenter.huaweicloud.com/d-3*****6/portal",
    "ssoRegion": "ap-southeast-1"
  },
  "ssoAuth": {
    "accessTokenResult": {
      "accessToken": "****",
      "expiresAt": "2024-05-08 00:52:50"
    },
    "accountId": "0cb8*****c3a0",
    "agencyUrn": "iam::0cb8*****c3a0:agency:Sys*****_*****_dc*****e8"
  }
}
```

1.5 Querying the Profile List

1.5.1 Querying System Parameters in the Profile List

KooCLI supports multiple profiles. Run the following command to query system parameters in the profile list. The query result can be output in a table or in JSON or TSV format. By default, the query result is output in JSON format.

hcloud configure list

```
hcloud configure list
{
  "language": "cn",
  "offline": "false",
  "agreePrivacy": "true",
  "warning": "true",
  "current": "test",
  "profiles": [
    {
      "name": "default",
      "mode": "AKSK",
      "accessKeyId": "*****",
      "secretAccessKey": "*****",
      "securityToken": "",
      "region": "ap-southeast-1",
      "projectId": "060576*****134588f135",
      "domainId": "",
      "skipSecureVerify": "false",
      "readTimeout": 10,
      "connectTimeout": 5,
      "retryCount": 0
    },
    {
      "name": "test",
      "mode": "AKSK",
      "accessKeyId": "*****",
      "secretAccessKey": "*****",
      "securityToken": "",
      "region": "ap-southeast-1",
      "projectId": "",
      "domainId": "",
      "skipSecureVerify": "false",
      "readTimeout": 10,
      "connectTimeout": 5,
      "retryCount": 0
    }
  ]
}
```

The values of sensitive parameters are **anonymized**.

Use **--cli-query** to filter the query result, and use **--cli-output** to specify the output format. The default output format is JSON. For example, output the query result in JSON format and filter the AK/SK in each profile:

```
hcloud configure list --cli-output=json --cli-query="profiles[]."
{Name:name,AK:accessKeyId,SK:secretAccessKey}"
[
  {
    "AK": "*****",
    "Name": "default",
    "SK": "*****"
  },
  {
    "AK": "*****",
    "Name": "test",
    "SK": "*****"
  }
]
```

To output the query result in a table, add **--cli-output=table** in the command.

```
hcloud configure list --cli-output=table --cli-query="profiles[]."
[name,accessKeyId,secretAccessKey,projectId]"
```

```
C:\cli>hcloud configure list --cli-output=table --cli-query="profiles[0]. [name, accessKeyId, secretAccessKey, projectId]"
+-----+-----+-----+-----+
| default | H9N****MXW | **** | 060576f135 |
+-----+-----+-----+-----+
| test    | 8NV****IOV | **** |              |
+-----+-----+-----+-----+
C:\cli>_
```

To output the query result in TSV format, add **--cli-output=tsv** in the command.

```
hcloud configure list --cli-output=tsv --cli-query="profiles[0].
[name,accessKeyId,secretAccessKey,projectId]"
```

```
C:\cli>hcloud configure list --cli-output=tsv --cli-query="profiles[0]. [name, accessKeyId, secretAccessKey, projectId]"
default H9N****MXW      ****      060576f135
C:\cli>_
```

1.5.2 Querying Custom Parameters in the Profile List

Run the following command to query custom parameters in the profile list of KooCLI. The query result can be output in a table or in JSON or TSV format. By default, the query result is output in JSON format.

hcloud configure list --cli-custom=true

```
hcloud configure list --cli-custom=true
{
  "current": "test",
  "profiles": [
    {
      "name": "default",
      "custom": {}
    },
    {
      "name": "test",
      "custom": {
        "password": {
          "value": "*****",
          "isEncrypted": true
        },
        "projectId": {
          "value": "068100000000000000000000f89d2e",
          "isEncrypted": false
        }
      }
    }
  ]
}
```

Encrypted custom parameter values are **anonymized**.

Use **--cli-query** to filter the query result, and use **--cli-output** to specify the output format. The default output format is JSON. In the following example, all custom parameters of a profile named **test** are queried, and the query result is output in JSON format:

```
hcloud configure list --cli-custom=true --cli-query="profiles[?name=='test'].custom" --cli-output=json
[
  {
    "password": {
      "isEncrypted": true,
      "value": "*****"
    },
    "projectId": {
```

```
    "isEncrypted": false,  
    "value": "06810000000000000000000000f89d2e"  
  }  
}
```

1.6 Querying a Profile

1.6.1 Querying System Parameters in a Profile

Use **--cli-profile** to specify the name of the profile to be queried. If no profile is specified, the **default profile** is queried. The query result can be output in a table or in JSON or TSV format. By default, the query result is output in JSON format. The command for querying system parameters in a profile is as follows:

hcloud configure show [--cli-profile=\${profileName}]

```
hcloud configure show --cli-profile=test  
{  
  "name": "test",  
  "mode": "AKSK",  
  "accessKeyId": "*****",  
  "secretAccessKey": "*****",  
  "securityToken": "",  
  "region": "ap-southeast-1",  
  "projectId": "",  
  "domainId": "",  
  "skipSecureVerify": "false",  
  "readTimeout": 10,  
  "connectTimeout": 5,  
  "retryCount": 0  
}
```

1.6.2 Querying Custom Parameters in a Profile

Use **--cli-profile** to specify the name of the profile to be queried. The query result can be output in a table or in JSON or TSV format. By default, the query result is output in JSON format. The command for querying custom parameters in a profile is as follows:

hcloud configure show [--cli-profile=\${profileName}] --cli-custom=true

```
hcloud configure show --cli-profile=test --cli-custom=true  
{  
  "password": {  
    "value": "*****",  
    "isEncrypted": true  
  },  
  "projectId": {  
    "value": "06810000000000000000000000f89d2e",  
    "isEncrypted": false  
  }  
}
```

1.7 Deleting a Profile

1.7.1 Deleting a Profile and Its Custom Parameters

Use **--cli-profile** to specify the name of the profile to be deleted. The command for deleting a profile is as follows:

hcloud configure delete --cli-profile=\${profileName}

```
hcloud configure delete --cli-profile=test
Profile test deleted. The default profile is default.
```

Deleting a profile will also delete its system parameters and custom parameters.

1.7.2 Deleting Custom Parameters in a Profile

Use **--cli-profile** to specify the name of the profile to be deleted. The command for deleting custom parameters in a profile is as follows:

- Deleting a **specific** custom parameter in a profile

**hcloud configure delete --cli-profile=\${profileName} [--cli-custom=true]
--cli-custom-key=\${customName}**

```
hcloud configure delete --cli-profile=test --cli-custom-key=projectId
From profile test, custom parameter projectId is deleted.
```

- Deleting **all** custom parameters in a profile

hcloud configure delete --cli-profile=\${profileName} --cli-custom=true

```
hcloud configure delete --cli-profile=test --cli-custom=true
All custom parameters in profile test are deleted.
```

1.8 Checking the Configuration File Format

To check the format of your local configuration file, run the following command:

hcloud configure test

```
hcloud configure test
The configuration file has the correct format.
```

1.9 Deleting All Profiles

Run the following command to delete all profiles:

hcloud configure clear

```
hcloud configure clear
Delete all configuration information? (y/N): y
All configuration information deleted.
```

2 HTTP Proxy

2.1 Configuring the HTTP Proxy

When you use KooCLI to manage and access cloud resources, cloud service APIs may fail to be called due to the intranet access restriction of your company. In this case, you can use the HTTP proxy by configuring the environment variables **http_proxy** and **https_proxy**.

Configuring the Environment Variable **http_proxy**

Run the command that matches your OS, to configure the environment variable **http_proxy**.

- Linux or macOS

- Temporary environment variable

Run the following command to configure **http_proxy**:

```
export http_proxy="http://username:password@proxyServer:port"
```

In the preceding command, **proxyServer** indicates the domain name (if resolvable) or IP address of the proxy server, and **port** indicates the port number. Enter a username and password if required, or run the following command:

```
export http_proxy="http://proxyServer:port"
```

- Permanent environment variable

To permanently configure **http_proxy** in an environment variable that takes effect only for the current user, add the following content to the **~/.bashrc** file:

```
export http_proxy="http://username:password@proxyServer:port"
```

In the preceding command, **proxyServer** indicates the domain name (if resolvable) or IP address of the proxy server, and **port** indicates the port number. Enter a username and password if required, or run the following command:

```
export http_proxy="http://proxyServer:port"
```

After configuring the permanent environment variable, log out and log in again for the variable to take effect.

- Windows
 - Using Command Prompt
 - Temporary environment variable

Run the following command to configure **http_proxy**:

```
set http_proxy=http://username:password@proxyServer:port
```

In the preceding command, **proxyServer** indicates the domain name (if resolvable) or IP address of the proxy server, and **port** indicates the port number. Enter a username and password if required, or run the following command:

```
set http_proxy=http://proxyServer:port
```
 - Permanent environment variable

Run the following command to permanently configure **http_proxy** in an environment variable that takes effect only for the current user:

```
setx "http_proxy" "http://username:password@proxyServer:port"
```

In the preceding command, **proxyServer** indicates the domain name (if resolvable) or IP address of the proxy server, and **port** indicates the port number. Enter a username and password if required, or run the following command:

```
setx "http_proxy" "http://proxyServer:port"
```

The permanent environment variable will take effect on the clients opened later.
 - Using PowerShell
 - Temporary environment variable

Run the following command to configure **http_proxy**:

```
$env:http_proxy="http://username:password@proxyServer:port"
```

In the preceding command, **proxyServer** indicates the domain name (if resolvable) or IP address of the proxy server, and **port** indicates the port number. Enter a username and password if required, or run the following command:

```
$env:http_proxy="http://proxyServer:port"
```
 - Permanent environment variable

Run the following command to permanently configure **http_proxy** in an environment variable that takes effect only for the current user:

```
[environment]::SetEnvironmentvariable("http_proxy", "http://username:password@proxyServer:port", "User")
```

In the preceding command, **proxyServer** indicates the domain name (if resolvable) or IP address of the proxy server, and **port** indicates the port number. Enter a username and password if required, or run the following command:

```
[environment]::SetEnvironmentvariable("http_proxy", "http://proxyServer:port", "User")
```

The permanent environment variable will take effect on the clients opened later.

Configuring the Environment Variable **https_proxy**

Run the command that matches your OS, to configure the environment variable **https_proxy**.

- Linux or macOS
 - Temporary environment variable

Run the following command to configure **https_proxy**:

```
export https_proxy="https://username:password@proxyServer:port"
```

In the preceding command, **proxyServer** indicates the domain name (if resolvable) or IP address of the proxy server, and **port** indicates the port number. Enter a username and password if required, or run the following command:

```
export https_proxy="https://proxyServer:port"
```
 - Permanent environment variable

To permanently configure **https_proxy** in an environment variable that takes effect only for the current user, add the following content to the **~/.bashrc** file:

```
export https_proxy="https://username:password@proxyServer:port"
```

In the preceding command, **proxyServer** indicates the domain name (if resolvable) or IP address of the proxy server, and **port** indicates the port number. Enter a username and password if required, or run the following command:

```
export https_proxy="https://proxyServer:port"
```

After configuring the permanent environment variable, log out and log in again for the variable to take effect.
- Windows
 - Using Command Prompt
 - Temporary environment variable

Run the following command to configure **https_proxy**:

```
set https_proxy=https://username:password@proxyServer:port
```

In the preceding command, **proxyServer** indicates the domain name (if resolvable) or IP address of the proxy server, and **port** indicates the port number. Enter a username and password if required, or run the following command:

```
set https_proxy=https://proxyServer:port
```
 - Permanent environment variable

Run the following command to permanently configure **https_proxy** in an environment variable that takes effect only for the current user:

```
setx "https_proxy" "https://username:password@proxyServer:port"
```

In the preceding command, **proxyServer** indicates the domain name (if resolvable) or IP address of the proxy server, and **port** indicates the port number. Enter a username and password if required, or run the following command:

```
setx "https_proxy" "https://proxyServer:port"
```

The permanent environment variable will take effect on the clients opened later.
 - Using PowerShell
 - Temporary environment variable

Run the following command to configure **https_proxy**:

```
$env:https_proxy="https://username:password@proxyServer:port"
```

In the preceding command, **proxyServer** indicates the domain name (if resolvable) or IP address of the proxy server, and **port** indicates

the port number. Enter a username and password if required, or run the following command:

```
$env:https_proxy="https://proxyServer:port"
```

- Permanent environment variable

Run the following command to permanently configure **https_proxy** in an environment variable that takes effect only for the current user:

```
[environment]::SetEnvironmentvariable("https_proxy", "https://  
username:password@proxyServer:port", "User")
```

In the preceding command, **proxyServer** indicates the domain name (if resolvable) or IP address of the proxy server, and **port** indicates the port number. Enter a username and password if required, or run the following command:

```
[environment]::SetEnvironmentvariable("https_proxy", "https://proxyServer:port", "User")
```

The permanent environment variable will take effect on the clients opened later.

2.2 Resolving the x509 Certificate Error

When you use KooCLI to manage and access cloud resources after configuring the HTTP proxy, the x509 error may occur when you call a cloud service API. The reason is that your proxy has modified the root certificate issued by the HTTPS website. As a result, the certificate becomes invalid when you access HTTPS resources on external networks.

To resolve this problem, import the certificate issued by your company to the trusted CA certificate of the OS. Alternatively, add **--cli-skip-secure-verify=true** to your command to **skip certificate verification** (not recommended).

3 Commonly Used Commands

3.1 Viewing Help Information

Add the **--help** parameter to a command to view help information. For example:

- `hcloud --help`
View the instructions for using KooCLI and the supported cloud services.
- `hcloud <service> --help`
View the operation list of a cloud service. For example, to view the operation list of ECS, run **hcloud ECS --help**.
- `hcloud <service> <operation> --help`
View details about a cloud service API. For example, to view details about the ECS API with operation **ShowServer**, run **hcloud ECS ShowServer --help**.

3.2 Turning On/Off Autocomplete

After you turn on autocomplete for KooCLI, if you press **Tab** when entering a command, a prompt will appear to help you autocomplete the command.

To turn on autocomplete, run the following command:

hcloud auto-complete on

hcloud auto-complete on
Autocomplete (bash only) turned on. If it does not work, run the ``bash`` command.

To turn off autocomplete, run the following command:

hcloud auto-complete off

hcloud auto-complete off
Autocomplete turned off. If it does not work, run the ``bash`` command.

NOTE

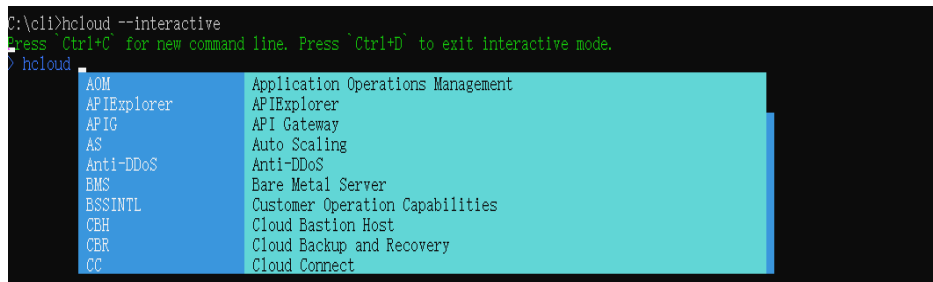
- Autocomplete is supported only in Bash.
- If the prompted parameter name contains **[N]**, which indicates an index, replace it with a number. If the prompted parameter name contains **{*}**, which indicates a custom parameter name, replace it with a string that does not contain periods (.).

3.3 Using Interactive Prompts

KooCLI provides powerful interactive prompts to facilitate command building on different platforms. Add **--interactive** in a command to enter the interactive mode. In this mode, the system displays cloud service names, API operations, parameters, and descriptions based on the information you enter. In addition, the system autocompletes the values of some parameters.

- Use the interactive mode as follows:
 - Use the **Tab** key or the up and down arrows to select the content to be autocompleted.
 - Press the space bar to add the selected content to the current command.
 - After entering the interactive mode, press **Ctrl+C** to switch to a new command line. To exit the interactive mode, press **Ctrl+D**.
- The following contents will be prompted and autocompleted in the interactive mode:
 - Cloud services
When you enter the interactive mode for the first time, the list of cloud services and system commands is automatically displayed.

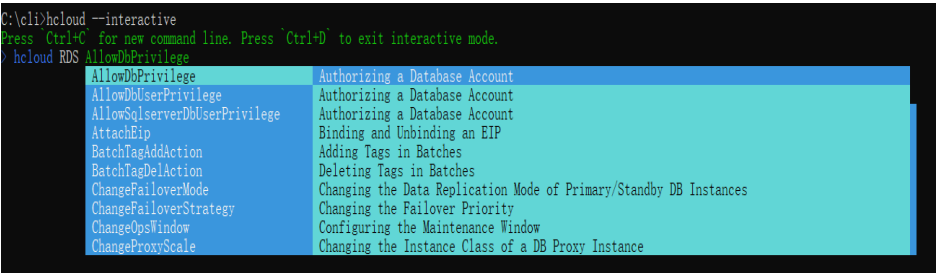
Figure 3-1 List of cloud services and system commands



In the cloud service list, the left column displays cloud service abbreviations or system commands, and the right column displays the full service names or system command descriptions.

- APIs
After you enter a valid cloud service name or system command, the API list of the service or the subcommands (or parameters) of the system command are displayed.
In the system command list, the left column displays the subcommands (or parameters) of the system command, and the right column displays the descriptions of the subcommands (or parameters).

Figure 3-2 Cloud service API list

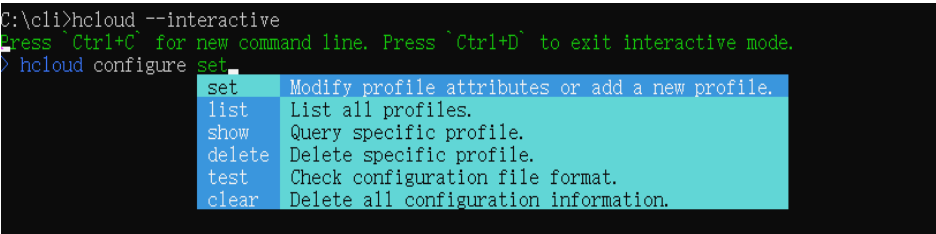


```
C:\cli>hcloud --interactive
Press Ctrl+C for new command line. Press Ctrl+D to exit interactive mode.
> hcloud RDS AllowDbPrivilege
```

AllowDbPrivilege	Authorizing a Database Account
AllowDbUserPrivilege	Authorizing a Database Account
AllowSqlServerDbUserPrivilege	Authorizing a Database Account
AttachEip	Binding and Unbinding an EIP
BatchTagAddAction	Adding Tags in Batches
BatchTagDelAction	Deleting Tags in Batches
ChangeFailoverMode	Changing the Data Replication Mode of Primary/Standby DB Instances
ChangeFailoverStrategy	Changing the Failover Priority
ChangeOpsWindow	Configuring the Maintenance Window
ChangeProxyScale	Changing the Instance Class of a DB Proxy Instance

In the cloud service API list, the left column displays API operation names, and the right column displays API descriptions.

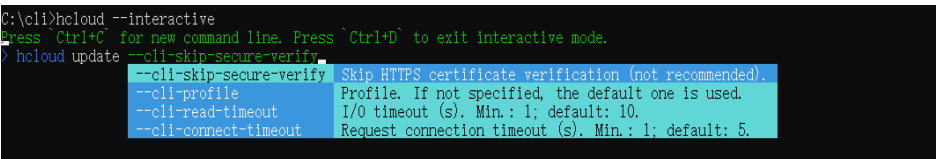
Figure 3-3 Subcommands of a system command



```
C:\cli>hcloud --interactive
Press Ctrl+C for new command line. Press Ctrl+D to exit interactive mode.
> hcloud configure set
```

set	Modify profile attributes or add a new profile.
list	List all profiles.
show	Query specific profile.
delete	Delete specific profile.
test	Check configuration file format.
clear	Delete all configuration information.

Figure 3-4 System command parameters



```
C:\cli>hcloud --interactive
Press Ctrl+C for new command line. Press Ctrl+D to exit interactive mode.
> hcloud update
```

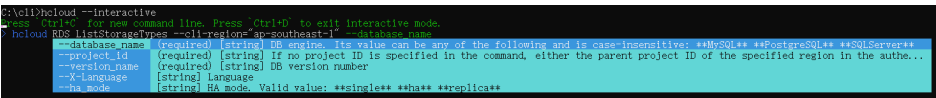
--cli-skip-secure-verify	Skip HTTPS certificate verification (not recommended).
--cli-profile	Profile. If not specified, the default one is used.
--cli-read-timeout	I/O timeout (s). Min.: 1; default: 10.
--cli-connect-timeout	Request connection timeout (s). Min.: 1; default: 5.

If the entered cloud service name or system command is invalid, no prompt is displayed.

- Parameters

After you enter a valid cloud service name and API operation, or a valid system command and subcommand, the parameter list of the API or subcommand is displayed.

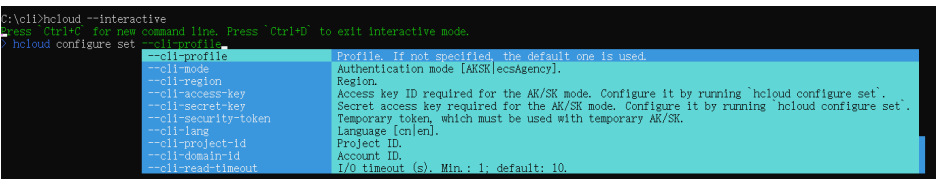
Figure 3-5 Cloud service API parameter list



```
C:\cli>hcloud --interactive
Press Ctrl+C for new command line. Press Ctrl+D to exit interactive mode.
> hcloud RDS listInstanceTypes
```

--database-name	(required) [string] DB engine. Its value can be any of the following and is case-insensitive: **MySQL** **PostgreSQL** **SQLServer**
--project-id	(required) [string] If no project ID is specified in the command, either the parent project ID of the specified region in the authentication file is used, or the default project ID is used.
--version-number	(required) [string] DB version number
--x-language	[string] Language
--ha-mode	[string] HA mode. Valid value: **single** **ha** **replica**

Figure 3-6 Subcommand parameter list of a system command



```
C:\cli>hcloud --interactive
Press Ctrl+C for new command line. Press Ctrl+D to exit interactive mode.
> hcloud configure set
```

--cli-profile	Profile. If not specified, the default one is used.
--cli-mode	Authentication mode [AKSK ecsAgency].
--cli-region	Region.
--cli-access-key	Access key ID required for the AK/SK mode. Configure it by running 'hcloud configure set'.
--cli-secret-key	Secret access key required for the AK/SK mode. Configure it by running 'hcloud configure set'.
--cli-security-token	Temporary token, which must be used with temporary AK/SK.
--cli-lang	Language [cn en].
--cli-project-id	Project ID.
--cli-account-id	Account ID.
--cli-read-timeout	I/O timeout (s). Min.: 1; default: 10.

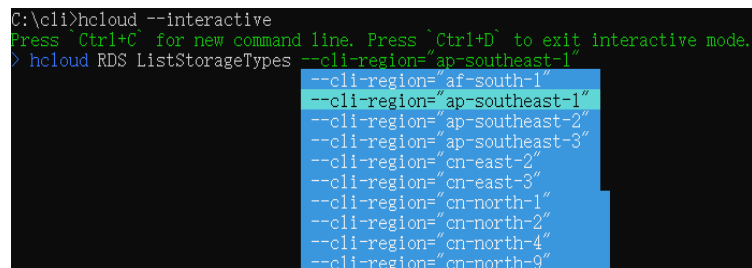
In the parameter list, the left column displays parameter names, and the right column displays their descriptions.

If the entered API operation or subcommand is invalid, or the entered parameter name is invalid, no prompt is displayed.

- Values

In interactive mode, if you enter an equal sign (=) after a parameter name, the allowed values or the default value of the parameter is displayed. This value prompt is available only for certain parameters. For example, if you enter `--cli-region=` in an API calling command, the regions where the API is available are displayed.

Figure 3-7 Regions where an API is available



```
C:\cli>hcloud --interactive
Press `Ctrl+C` for new command line. Press `Ctrl+D` to exit interactive mode.
> hcloud RDS ListStorageTypes --cli-region="ap-southeast-1"
--cli-region="af-south-1"
--cli-region="ap-southeast-1"
--cli-region="ap-southeast-2"
--cli-region="ap-southeast-3"
--cli-region="cn-east-2"
--cli-region="cn-east-3"
--cli-region="cn-north-1"
--cli-region="cn-north-2"
--cli-region="cn-north-4"
--cli-region="cn-north-9"
```

- Shortcut keys in interactive mode
 - **Ctrl+W**: Delete the word before the cursor.
 - **Ctrl+K**: Delete the content after the cursor.
 - **Ctrl+U**: Delete the content before the cursor.
 - **Ctrl+L**: Clear the screen.

NOTE

- If no profile is added, or the region specified in the **default profile** is not supported by the API in the command, select a region from the **cli-region** list prompted for the API. Then, KooCLI continues to display the parameter list of the API.
- In interactive prompts, parameters except custom map type parameters (containing **{*}** in their names) will not be prompted again once entered. If a parameter name contains **[N]**, which indicates an index, replace the character with a number. If a parameter name contains **{*}**, which indicates a custom parameter name, replace the character with a string that does not contain periods (.).
- After switching to a new command line, you can use the up and down arrows to browse the records of executed commands.

3.4 Managing Metadata

- Clearing metadata caches

KooCLI caches local metadata files obtained during API calling and stores the files in the following directory:

- Online mode

- Windows: `C:\Users\{Your Windows username}\.hcloud\metaRepo\`
- Linux: `/home/{current-username}/.hcloud/metaRepo/`

- macOS: `/Users/{Current username}/.hcloud/metaRepo/`
- Offline mode
 - Windows: `C:\Users\{Your Windows username}\.hcloud\metaOrigin\`
 - Linux: `/home/{Current username}/.hcloud/metaOrigin/`
 - macOS: `/Users/{Current username}/.hcloud/metaOrigin/`

Run the following commands to clear metadata cache files:

- Online mode

hcloud meta clear

```
hcloud meta clear
Cache cleared.
```

- Offline mode

Run the **hcloud meta clear** command to clear the metadata cache files parsed from the downloaded offline metadata package. The package will remain. During API calling, this package will be parsed again and new metadata cache files will be written into it. To completely delete the offline metadata package and the contained metadata cache files, delete the **directory** where these files are located, according to the OS you use.

- Downloading metadata

A downloaded offline metadata package will be stored in the preceding **directories for the offline mode**. Run the following command to download the offline metadata package:

hcloud meta download

```
hcloud meta download
Download successful.
```

3.5 Querying the Current Version

To query the version of KooCLI, run the following command:

hcloud version

```
hcloud version
Current KooCLI version: 3.2.8
```

3.6 Upgrading the Version

KooCLI supports local upgrade. Run the following command to upgrade it to the latest version:

hcloud update

```
hcloud update
KooCLI will be upgraded to the latest version. Continue? (y/N): y
Upgrade successful.
```

To skip the confirmation for entering the interactive mode, add **-y** to the command.

```
hcloud update -y
Upgrade successful.
```

3.7 Managing Logs

KooCLI provides log recording and management to cache the logs generated during API calling. Log files are stored in the following directories:

- Windows: `C:\Users\{Your Windows username}\hcloud\log\`
- Linux: `/home/{Current username}/hcloud/log/`
- macOS: `/Users/{Current username}/hcloud/log/`

Parameters related to log management:

- **level:** log level, which can be **info**, **warning**, or **error**.
- **max-file-size:** maximum size (MB) of a single log file. Range: 1 to 100. Default value: **20**.
- **max-file-num:** the number of retained log files. The value **0** indicates that all log files are retained. Default value: **3**.
- **retention-period:** the number of days for retaining log files. The value **0** indicates that log files are retained permanently.

If none of the parameters are specified, the default log level is error, a single log file is 20 MB, and 3 files are retained.

Run the following command to set log-related parameters:

hcloud log set --key1=value1 --key2=value2 ...

```
hcloud log set --level=error --max-file-size=20 --max-file-num=3 --retention-period=30
Configuration successful.
```

Run the following command to view log-related parameters:

hcloud log show

```
hcloud log show
{
  "maxFileSize": 20,
  "maxFileNum": 3,
  "logLevel": "error",
  "logRetentionPeriod": 30
}
```

3.8 Managing Data in OBS

KooCLI has integrated the obsutil tool of Object Storage Service (OBS). You can run **hcloud obs** to manage your data in OBS.

Function Overview

With KooCLI, you can perform the following operations to manage your data in OBS.

Table 3-1 Integrated OBS functions in KooCLI

Function	Description
Basic bucket operations	Create buckets of different storage classes in specific regions, delete buckets, and obtain the bucket list and configuration information.
Basic object operations	Manage objects, including uploading, downloading, deleting, and listing objects. • Upload one or more files or folders. • Upload large files in multiple parts. • Synchronously upload, download, and copy incremental objects. • Copy a single object or copy multiple objects by name prefix. • Move a single object or move multiple objects by name prefix. • Resume failed upload, download, or copy tasks.
Logs	Configure logging on your client to record bucket and object operations for statistical analysis.

Initial Configuration

Before using KooCLI to manage data in OBS, configure OBS connection, including the OBS endpoint and access key (AK/SK). You can use KooCLI to manage OBS buckets and objects only after being authenticated by OBS.

- **Permanent AK/SK:**
`hcloud obs config -i=ak -k=sk -e=endpoint`
- **Temporary AK/SK and SecurityToken:**
`hcloud obs config -i=ak -k=sk -t=token -e=endpoint`

Checking Connectivity

After the configuration is complete, run the following command to check the connectivity:

```
hcloud obs ls -s
```

Check the configuration result based on the command output:

- **Bucket number:** The configuration is correct.
- **Http status [403]:** The access key is incorrect, or you do not have permission to access the bucket list. Analyze the root cause.
- **A connection attempt failed:** OBS cannot be connected. Check the network condition.

Command Line Structure

The commands for managing OBS data with KooCLI have the following structure:

```
hcloud obs command [parameters...] [options...]
```

To enter the interactive mode in Windows:

Step 1 Run the following command to enter the interactive mode:

```
hcloud obs
```

Step 2 Manage your OBS data with commands in the following structure:

```
command [parameters...] [options...]
```

For example:

```
hcloud obs
Enter "exit" or "quit" to logout
Enter "help" or "help command" to show help docs
Input your command:
-->ls -limit=3 -s
obs://bucket-001
obs://bucket-002
obs://bucket-003
Bucket number: 3
```

----End

NOTE

- **command** indicates the command to be executed, for example, **ls** or **cp**.
- **parameters** indicates the basic parameters (mandatory) of the command, for example, name of a bucket to be created.
- **options** indicates additional parameters (optional) of the command. Additional parameters must be preceded with a hyphen (-) when you run the command. Enter a parameter in the *-key=value* or *-key value* format, for example, **-acl=private** or **-acl private**. There is no difference between the two formats. Select either one as you like.
- The brackets ([]) are not part of the command. Do not enclose parameter values with brackets when entering a command.
- If the command contains special characters such as ampersands (&), angle brackets (<>), and spaces, escape these parameters using single quotation marks (Linux and macOS) or quotation marks (Windows).

NOTE

When using KooCLI to run an OBS command on a delegated ECS, add the **-authSource=ecsAgency** parameter to the end of the command to automatically obtain temporary authentication information based on the ECS agency.

The following table lists the OBS operation commands supported by KooCLI. The parameters of each command are the same as those in obsutil. For details about these parameters, see [Bucket Commands](#), [Object Commands](#), and [Auxiliary Commands](#).

Table 3-2 OBS operation commands supported by KooCLI

Cat eg ory	Co m m a n d	Fu n c t i o n	Description	Command Line Structure
Bu cke t co m ma nds	mb	Cr ea te bu ck et	Create a bucket with a unique name. Each account can create a maximum of 100 buckets.	hcloud obs mb obs://bucket [-fs] [-az=xxx] [-acl=xxx] [-sc=xxx] [-location=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]
	ls	Lis t bu ck et s	Obtain the bucket list. The buckets are displayed in lexicographical order of their names.	hcloud obs ls [-s] [-du] [-sc] [-j=1] [-limit=1] [-format=default] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]
	stat	Q ue ry bu ck et pr op ert ies	Query the basic properties of a bucket, including the default storage class, region, version, support for POSIX, AZ, number of objects, storage usage, and bucket quota.	hcloud obs stat obs://bucket [-acl] [-bf=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]
	chattri	Se t bu ck et pr op ert ies	Set the properties of a bucket, including storage classes and access policies.	hcloud obs chattri obs://bucket [-sc=xxx] [-acl=xxx] [-aclXml=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]
	rm	De le te bu ck et	Delete a bucket that contains no objects, historical versions, or fragments.	hcloud obs rm obs://bucket [-f] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]

Cat eg ory	Co m m a n d	Fu n c t i o n	Description	Command Line Structure
	bu ck et po lic y	Co nfi gu re bu ck et po lic y	Configure a bucket policy.	hcloud obs bucketpolicy obs://bucket -method=put -localfile=xxx [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]
		O bt ai n bu ck et po lic y	Obtain a bucket policy.	hcloud obs bucketpolicy obs://bucket -method=get [-localfile=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]
		De let e bu ck et po lic y	Delete a bucket policy.	hcloud obs bucketpolicy obs://bucket -method=delete [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]
Obj ect co m ma nds	m kd ir	Cr ea te fol de r	Create a folder in a specified bucket or the local file system.	- Create a folder in a specified bucket. hcloud obs mkdir obs://bucket/folder[/subfolder1/subfolder2] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx] - Create a folder in the local file system. hcloud obs mkdir folder_url [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]

Cat eg ory	Co m m a n d	Fu n c t i o n	Description	Command Line Structure
	cp	U p l o a d o b j e c t	Upload one or more local files or folders to a specified path in OBS. These files can be texts, images, or videos.	• Upload a file. hcloud obs cp file_url obs://bucket[/key] [-arcDir=xxx] [-dryRun] [-link] [-u] [-vlength] [-vmd5] [-p=1] [-threshold=5248800] [-acl=xxx] [-sc=xxx] [-meta=aaa:bbb#ccc:ddd] [-ps=auto] [-o=xxx] [-cpd=xxx] [-fr] [-o=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx] • Upload a folder. hcloud obs cp folder_url obs://bucket[/key] -r [-arcDir=xxx] [-dryRun] [-link] [-f] [-flat] [-u] [-vlength] [-vmd5] [-j=1] [-p=1] [-threshold=52428800] [-acl=xxx] [-sc=xxx] [-meta=aaa:bbb#ccc:ddd] [-ps=auto] [-include=*.xxx] [-exclude=*.xxx] [-timeRange=time1-time2] [-mf] [-o=xxx] [-cpd=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx] • Upload multiple files and folders. hcloud obs cp file1_url,folder1_url filelist_url obs://bucket[/prefix] -msm=1 [-r] [-arcDir=xxx] [-dryRun] [-link] [-f] [-u] [-vlength] [-vmd5] [-flat] [-j=1] [-p=1] [-threshold=52428800] [-acl=xxx] [-sc=xxx] [-meta=aaa:bbb#ccc:ddd] [-ps=auto] [-include=*.xxx] [-exclude=*.xxx] [-timeRange=time1-time2] [-at] [-mf] [-o=xxx] [-cpd=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]
		Co p y o b j e c t	Copy a single object or copy multiple objects with a specified name prefix.	• Copy a single object. hcloud obs cp obs://srcbucket/key obs://dstbucket/[dest] [-dryRun] [-u] [-crr] [-vlength] [-vmd5] [-p=1] [-threshold=52428800] [-versionId=xxx] [-acl=xxx] [-sc=xxx] [-meta=aaa:bbb#ccc:ddd] [-ps=auto] [-cpd=xxx] [-fr] [-o=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx] • Copy multiple objects. hcloud obs cp obs://srcbucket[/key] obs://dstbucket[/dest] -r [-dryRun] [-f] [-flat] [-u] [-crr] [-vlength] [-vmd5] [-j=1] [-p=1] [-threshold=52428800] [-acl=xxx] [-sc=xxx] [-meta=aaa:bbb#ccc:ddd] [-ps=auto] [-include=*.xxx] [-exclude=*.xxx] [-timeRange=time1-time2] [-mf] [-o=xxx] [-cpd=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]
		D o w n l o a d o b j e c t	Download an object or download multiple objects with a specified name prefix to your local PC.	• Download a single object. hcloud obs cp obs://bucket/key file_or_folder_url [-tempFileDir=xxx] [-dryRun] [-u] [-vlength] [-vmd5] [-p=1] [-threshold=52428800] [-versionId=xxx] [-ps=auto] [-cpd=xxx] [-fr] [-o=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx] • Download multiple objects. hcloud obs cp obs://bucket[/key] folder_url -r [-tempFileDir=xxx] [-dryRun] [-f] [-flat] [-u] [-vlength] [-vmd5] [-j=1] [-p=1] [-threshold=52428800] [-ps=auto] [-include=*.xxx] [-exclude=*.xxx] [-timeRange=time1-time2] [-mf] [-o=xxx] [-cpd=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]

Cat eg ory	Co m m a n d	Fu n c t i o n	Description	Command Line Structure
		Re su m e fai le d up lo ad ta sk	Resume a failed upload task based on the task ID.	<code>hcloud obs cp -recover=xxx [-arcDir=xxx] [-dryRun] [-f] [-u] [-vlength] [-vmd5] [-j=1] [-p=1] [-threshold=52428800] [-acl=xxx] [-sc=xxx] [-meta=aaa:bbb#ccc:ddd] [-ps=auto] [-include=*.xxx] [-exclude=*.xxx] [-timeRange=time1-time2] [-mf] [-o=xxx] [-cpd=xxx] [-clear] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]</code>
		Re su m e fai le d co py ta sk	Resume a failed copy task based on the task ID.	<code>hcloud obs cp -recover=xxx [-dryRun] [-f] [-u] [-crr] [-vlength] [-vmd5] [-j=1] [-p=1] [-threshold=52428800] [-acl=xxx] [-sc=xxx] [-meta=aaa:bbb#ccc:ddd] [-ps=auto] [-include=*.xxx] [-exclude=*.xxx] [-timeRange=time1-time2] [-mf] [-o=xxx] [-cpd=xxx] [-clear] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]</code>
		Re su m e fai le d do wn lo ad ta sk	Resume a failed download task based on the task ID.	<code>hcloud obs cp -recover=xxx [-dryRun] [-tempFileDir=xxx] [-f] [-u] [-vlength] [-vmd5] [-j=1] [-p=1] [-threshold=52428800] [-ps=auto] [-include=*.xxx] [-exclude=*.xxx] [-timeRange=time1-time2] [-mf] [-o=xxx] [-cpd=xxx] [-clear] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]</code>

Cat eg ory	Co m m a n d	Fu n c t i o n	Description	Command Line Structure
	st at	Q u e r y o b j e c t p r o p e r t i e s	Query the basic properties of an object.	hcloud obs stat obs://bucket/key [-acl] [-bf=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]
	ch att ri	S e t o b j e c t p r o p e r t i e s	Set properties of a single object or of multiple objects with a specified name prefix.	- Set properties of a single object. hcloud obs chattri obs://bucket/key [-sc=xxx] [-acl=xxx] [-aclXml=xxx] [-versionId=xxx] [-fr] [-o=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx] - Set properties of multiple objects. hcloud obs chattri obs://bucket[/key] -r [-f] [-v] [-sc=xxx] [-acl=xxx] [-aclXml=xxx] [-o=xxx] [-j=1] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]
	ls	L i s t o b j e c t s	Query objects or object versions in a bucket. All objects are displayed in lexicographical order of their names or versions.	hcloud obs ls obs://bucket[/prefix] [-s] [-d] [-v] [-du] [-marker=xxx] [-versionIdMarker=xxx] [-bf=xxx] [-limit=1] [-format=default] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]
		L i s t m u l t i p a r t u p l o a d t a s k s	Query multipart upload tasks in a bucket.	hcloud obs ls obs://bucket[/prefix] [-s] [-d] -m [-a] [-uploadIdMarker=xxx] [-marker=xxx] [-limit=1] [-format=default] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]

Cat eg ory	Co m m a n d	Fu n c t i o n	Description	Command Line Structure
	mv	Move object	Move a single object or move multiple objects with a specified name prefix.	● Move a single object. hcloud obs mv obs://srcbucket/key obs://dstbucket/[dest] [-dryRun] [-u] [-p=1] [-threshold=52428800] [-versionId=xxx] [-acl=xxx] [-sc=xxx] [-meta=aaa:bbb#ccc:ddd] [-ps=auto] [-cpd=xxx] [-fr] [-o=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx] ● Move multiple objects. hcloud obs mv obs://srcbucket/[key] obs://dstbucket/[dest] -r [-dryRun] [-f] [-flat] [-u] [-j=1] [-p=1] [-threshold=52428800] [-acl=xxx] [-sc=xxx] [-meta=aaa:bbb#ccc:ddd] [-ps=auto] [-include=*.xxx] [-exclude=*.xxx] [-timeRange=time1-time2] [-mf] [-o=xxx] [-cpd=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]
	sign	Generate object download link	Generate a download link for a specified object in a bucket or generate download links for multiple objects with a specified name prefix in a bucket.	● Generate a download link for a single object. hcloud obs sign obs://bucket/key [-e=300] [-config=xxx] [-endpoint=xxx] [-i=xxx] [-k=xxx] [-t=xxx] ● Generate download links for multiple objects with a specified name prefix. hcloud obs sign obs://bucket/[key] -r [-e=300] [-timeRange=time1-time2] [-include=*.xxx] [-exclude=*.xxx] [-o=xxx] [-config=xxx] [-endpoint=xxx] [-i=xxx] [-k=xxx] [-t=xxx]
	rm	Delete object	● Delete a specified object. ● Delete multiple objects with a specified name prefix.	● Delete a single object. hcloud obs rm obs://bucket/key [-f] [-versionId=xxx] [-fr] [-o=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx] ● Delete multiple objects. hcloud obs rm obs://bucket/[key] -r [-j=1] [-f] [-v] [-o=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]

Cat eg ory	Co m m a n d	Fu n c t i o n	Description	Command Line Structure
	sy nc	Sy nc h r o n o u s l y u p l o a d i n c r e m e n t a l o b j e c t s	Synchronize all content in a local source path to a specified target OBS bucket to keep data consistency. Incremental synchronization has the following meanings: 1. Increment: Compare the source files with the target objects and upload only the source files that have changed. 2. Synchronization : After the command is executed, ensure that the local source path is a subset of the target bucket on OBS. That is, any file in the local source path has its corresponding object in the target bucket on OBS.	• Upload a file synchronously. hcloud obs sync file_url obs://bucket[/key] [-arcDir=xxx] [-dryRun] [-link] [-vlength] [-vmd5] [-p=1] [-threshold=5248800] [-acl=xxx] [-sc=xxx] [-meta=aaa:bbb#ccc:ddd] [-ps=auto] [-o=xxx] [-cpd=xxx] [-fr] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx] • Upload a folder synchronously. hcloud obs sync folder_url obs://bucket[/key] [-arcDir=xxx] [-dryRun] [-link] [-vlength] [-vmd5] [-j=1] [-p=1] [-threshold=52428800] [-acl=xxx] [-sc=xxx] [-meta=aaa:bbb#ccc:ddd] [-ps=auto] [-include=*.*] [-exclude=*.*] [-timeRange=time1-time2] [-at] [-mf] [-o=xxx] [-cpd=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]

Cat eg ory	Co m m a n d	Fu n c t i o n	Description	Command Line Structure
		Sy nc h r o n i z e o u t s y c o p y i n c r e m e n t a l o b j e c t s	Synchronize all objects in a specified path of the source bucket to a specified path in the destination bucket to keep data consistency. Incremental synchronization has the following meanings: 1. Increment: Compare the source and target objects and copy only the source objects that have changed. 2. Synchronization: After the command is executed, ensure that the specified path of the source bucket is a subset of the specified path in the target bucket. That is, any object in the specified path of the source bucket has its corresponding object in the target bucket.	<pre>hcloud obs sync obs://srcbucket[/key] obs://dstbucket[/dest] [-dryRun] [-crr] [-vlength] [-vmd5] [-j=1] [-p=1] [-threshold=52428800] [-acl=xxx] [-sc=xxx] [-meta=aaa:bbb#ccc:ddd] [-ps=auto] [-include=*.xxx] [-exclude=*.xxx] [-timeRange=time1-time2] [-mf] [-o=xxx] [-cpd=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]</pre>

Cat eg ory	Co m m a n d	Fu n c t i o n	Description	Command Line Structure
		Sy nc h r o n i z e o b j e c t s	Synchronize all objects in a specified path of the source OBS bucket to a target local path to keep data consistency. Incremental synchronization has the following meanings: 1. Increment: Compare the source objects with the target files and download only the source objects that have changed. 2. Synchronization: After the command is executed, ensure that the specified path of the source bucket is a subset of the target local path. That is, any object in the specified path of the source bucket has its corresponding file in the target local path.	<pre>hcloud obs sync obs://bucket[/key] folder_url [-tempFileDir=xxx] [-dryRun] [-vlength] [-vmd5] [-j=1] [-p=1] [-threshold=52428800] [-ps=auto] [-include=*.xxx] [-exclude=*.xxx] [-timeRange=time1-time2] [-mf] [-o=xxx] [-cpd=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]</pre>

Cat eg ory	Co m m a n d	Fu n c t i o n	Description	Command Line Structure
	re st ore	Re st ore o b j e c t s f r o m O B S a r c h i v e	Restore a specified object whose storage class is cold or restore multiple objects with a specified name prefix.	• Restore a single object. hcloud obs restore obs://bucket/key [-d=1] [-t=xxx] [-versionId=xxx] [-fr] [-o=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-token=xxx] • Restore multiple objects. hcloud obs restore obs://bucket[/key] -r [-f] [-v] [-d=1] [-t=xxx] [-o=xxx] [-j=1] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-token=xxx] • Restore all objects in a specified directory. hcloud obs restore obs://bucket/folder/ -r [-f] [-v] [-d=1] [-t=xxx] [-o=xxx] [-j=1] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-token=xxx]
	ab or t	De let e m u l t i p a r t u p l o a d t a s k	• Delete a multipart upload task of a specified ID in a specified bucket. • Delete multiple multipart upload tasks with a specified name prefix.	• Delete a single multipart upload task. hcloud obs abort obs://bucket/key -u=xxx [-f] [-fr] [-o=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx] • Delete multiple multipart upload tasks. hcloud obs abort obs://bucket[/key] -r [-f] [-o=xxx] [-j=1] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]

Cat eg ory	Co m m a n d	Fu n c t i o n	Description	Command Line Structure
	cr ea te- sh ar e	Cr ea te au th ori za tio n co de fo r dir ec to ry sh ari ng	Create an authorization code for directory sharing by specifying a bucket name, object name prefix, and access code.	<pre>hcloud obs create-share obs://bucket[/prefix] [-ac=xxx] [-vp=xxx] [-dst=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]</pre>
	sh ar e- ls	Lis t ob jec ts by usi ng au th ori za tio n co de	Query objects in a bucket with an authorization code. The returned objects are displayed in lexicographical order of their names.	• Enter an authorization code directly. <pre>hcloud obs share-ls authorization_code [-ac=xxx] [-prefix=xxx] [-s] [-d] [-marker=xxx] [-bf=xxx] [-limit=1] [-format=default] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]</pre> • Pass an authorization code through a file path. <pre>hcloud obs share-ls file://authorization_code_file_url [-ac=xxx] [-prefix=xxx] [-s] [-d] [-marker=xxx] [-bf=xxx] [-limit=1] [-format=default] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]</pre>

Cat eg ory	Co m m a n d	Fu n c t i o n	Description	Command Line Structure
	sh a r e - c p	D o w n l o a d o b j e c t s b y u s i n g a u t h o r i z a t i o n c o d e	Download a single object or download multiple objects with a specified name prefix to your local PC by using an authorization code.	- Download a single object by directly entering an authorization code. <pre>hcloud obs share-cp authorization_code file_or_folder_url -key=xxx [-ac=xxx] [-dryRun] [-tempFileDir=xxx] [-u] [-vlength] [-vmd5] [-p=1] [-threshold=52428800] [-ps=auto] [-cpd=xxx] [-fr] [-o=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]</pre> - Download a single object by passing an authorization code through a file path. <pre>hcloud obs share-cp file:// authorization_code_file_url file_or_folder_url -key=xxx [-ac=xxx] [-dryRun] [-tempFileDir=xxx] [-u] [-vlength] [-vmd5] [-p=1] [-threshold=52428800] [-ps=auto] [-cpd=xxx] [-fr] [-o=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]</pre> - Download multiple objects by directly entering an authorization code. <pre>hcloud obs share-cp authorization_code folder_url -r [-key=xxx] [-ac=xxx] [-dryRun] [-tempFileDir=xxx] [-f] [-u] [-vlength] [-vmd5] [-flat] [-j=1] [-p=1] [-threshold=52428800] [-ps=auto] [-include=*.xxx] [-exclude=*.xxx] [-timeRange=time1-time2] [-mf] [-o=xxx] [-cpd=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]</pre> - Download multiple objects by passing an authorization code through a file path. <pre>hcloud obs share-cp file:// authorization_code_file_url folder_url -r [-key=xxx] [-ac=xxx] [-dryRun] [-tempFileDir=xxx] [-f] [-u] [-vlength] [-vmd5] [-flat] [-j=1] [-p=1] [-threshold=52428800] [-ps=auto] [-include=*.xxx] [-exclude=*.xxx] [-timeRange=time1-time2] [-mf] [-o=xxx] [-cpd=xxx] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]</pre>
Au x i l i a r y c o m m a n d s	co n f i g	U p d a t e c o n f i g u r a t i o n f i l e	Update some items in the .obsutilconfig file, including the endpoint, AK, SK, and token. For details about the parameters in .obsutilconfig , see Parameter Description .	- Update configurations in interactive mode <pre>hcloud obs config [-interactive] [-crr] [-config=xxx]</pre> - Directly update configurations. <pre>hcloud obs config [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx] [-crr] [-config=xxx]</pre>

Cat eg ory	Co m m a n d	Fu n c t i o n	Description	Command Line Structure
	cle ar	De le t e p a r t r e c o r d s	Delete part records from a specified directory.	hcloud obs clear [checkpoint_dir] [-u] [-d] [-c] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]
	he lp	Vi e w c o m m a n d h e l p i n f o r m a t i o n	View the supported OBS commands or the help information of a specific command.	hcloud obs help [command]
	ve r s i o n	Vi e w v e r s i o n	View the version of the integrated obsutil.	hcloud obs version
	ar ch ive	Ar ch ive l o g f i l e s	Archive log files to a local PC or to a specified bucket.	- Archive to a local PC. hcloud obs archive [file_or_folder_url] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx] - Archive to a specified bucket. hcloud obs archive obs://bucket[/key] [-config=xxx] [-e=xxx] [-i=xxx] [-k=xxx] [-t=xxx]

Cat eg ory	Co m m a n d	Fu n c t i o n	Description	Command Line Structure
	ls	List failure result files	List the last modified failure result files generated by the cp or sync commands in a specified folder.	hcloud obs ls -failed [-limit=1000] [-o=xxx]

3.9 Managing CCE Clusters

KooCLI provides commands for managing CCE clusters. Run the commands listed below to manage your CCE clusters. For details, see [Accessing a Cluster Using KooCLI and kubectl](#).

Table 3-3 Managing CCE clusters using KooCLI

Command	Function
hcloud cce list-cluster	Show clusters of the current user.
hcloud cce update-kubeconfig	Generate kubeconfig for the target cluster.
hcloud cce token	Use an IAM credential to generate an authentication token in Kubernetes ExecCredential format.

4 Options

4.1 Option Overview

KooCLI options are system parameters that can be added to commands for calling APIs. The following table describes the options supported by KooCLI. Any option except **help**, **debug**, **dryrun**, **cli-output**, **cli-query**, **cli-output-num**, **cli-jsonInput**, and **cli-endpoint** can be set in profiles. When a command is executed, the value of a parameter in the command takes precedence over that in the profile.

Table 4-1 KooCLI options

Option	Description	Example
help	Prints help information.	hcloud RDS ListCollations --cli-region="ap-southeast-1" --help
debug	Prints debugging information during command invocation. For example, the steps of API calling and the complete request URL.	hcloud VPC ShowVpc/v3 --cli-region="ap-southeast-1" --project_id="0dd8cb*****19b5a84546" --vpc_id="0bbe****_****_****_****_****235be6e7" --debug
dryrun	Checks a command. The system prints the request message after verification. It does not call the target API.	hcloud RDS CreateConfiguration --cli-region="ap-southeast-1" --project_id="4ff018c*****df31948" --datastore.type="MySQL" --datastore.version="5.7" --values.max_connections="10" --name="test-001" --description="test create configuration" --dryrun
skeleton	Generates an API input parameter skeleton for passing API parameters using --cli-jsonInput .	hcloud RDS CreateConfiguration --cli-region="ap-southeast-1" --skeleton
cli-region	Region where the cloud service resources to be managed are located.	hcloud EVS DeleteVolume --cli-region="ap-southeast-1" --volume_id="aed9****_****_****_****_****0e3219cf" --project_id="0dd8cb*****19b5a84546"

Option	Description	Example
cli-access-key, cli-secret-key, cli-security-token	• cli-access-key: access key ID (AK). This parameter must be used together with SK. • cli-secret-key: secret access key (SK). This parameter must be used together with AK. • cli-security-token: temporary security credential. This parameter is required when you use a temporary AK/SK for identity authentication. These options are used to call cloud service APIs using an AK/SK in non-configuration mode.	Call cloud service APIs using an AK/SK in non-configuration mode. • Use an access key (permanent AK/SK): <code>hcloud RDS ListApiVersion --cli-region="ap-southeast-1" --cli-access-key=***** --cli-secret-key=*****</code> • Use temporary security credentials (temporary AK/SK and SecurityToken): <code>hcloud RDS ListApiVersion --cli-region="ap-southeast-1" --cli-access-key=***** --cli-secret-key=***** --cli-security-token=*****</code>
cli-agency-domain-id/cli-agency-domain-name, cli-agency-name, cli-source-profile	• cli-agency-domain-name: Account name of a delegating party. This parameter must be used together with cli-agency-name. • cli-agency-domain-id: Account ID of the delegating party. This parameter must be used together with cli-agency-name. • cli-agency-name: Agency name. It must be used together with cli-agency-domain-id or cli-agency-domain-name. • cli-source-profile: The profile that stores the authentication information of the delegated party. The value of this option cannot be the current profile.	<code>hcloud VPC ListAddressGroup/v3 --cli-region="ap-southeast-1" --project_id="2cc60*****caefa5019ef" --cli-agency-domain-id=13534326*****5cf67b --cli-agency-name=***** --cli-source-profile=test</code>

Option	Description	Example
cli-domain-id	ID of the account to which the IAM user belongs. This option is required when global service APIs are called in AK/SK authentication mode. In this case, KooCLI automatically obtains this parameter from your authentication information.	<pre>hcloud CDN ListDomains --cli-region="ap-southeast-1" --cli-domain-id="08e09a6e*****1bb800"</pre>
cli-profile	Name of a KooCLI profile. A profile stores a group of common information (such as the AK/SK, region, and project ID) required for calling cloud service APIs.	<pre>hcloud EVS ListSnapshots --cli-profile=test</pre>
cli-mode	Authentication mode of a profile. Options: - AKSK - ecsAgency - SSO - AssumeRole	- Set cli-mode to AKSK. <pre>hcloud CCE ListNodes --cluster_id="f288****_****_****_****_ac101534" --project_id="0dd8cb*****19b5a84546" --cli-profile=test --cli-mode=AKSK</pre> - Set cli-mode to ecsAgency. <pre>hcloud CCE ListNodes --cluster_id="f288****_****_****_****_ac101534" --project_id="0dd8cb*****19b5a84546" --cli-profile=test --cli-mode=ecsAgency</pre>
cli-output, cli-query, cli-output-num	Output format. - cli-output Output format of the response data. The options include: - json - table - tsv - cli-query JMESPath for filtering response data. - cli-output-num Indicates whether to print the row numbers during table output. The value can be true or false.	- When the value of cli-output is json: - Call a cloud service API <pre>hcloud CCE ListClusters --cli-region="ap-southeast-1" --type="VirtualMachine" --project_id="0dd8cb*****19b5a84546" --cli-query="items[0]"</pre> - Call a CLI system command <pre>hcloud configure list --cli-output=json --cli-query="profiles[]. {Name:name,Mode:mode,Ak:accessKeyId,SK:secretAccessKey}"</pre> - When the value of cli-output is table: <pre>hcloud configure list --cli-output=table --cli-query="profiles[]. {Name:name,Mode:mode,Ak:accessKeyId,SK:secretAccessKey}"</pre> - When the value of cli-output is tsv: <pre>hcloud configure list --cli-output=tsv --cli-query="profiles[]. {Name:name,Mode:mode,Ak:accessKeyId,SK:secretAccessKey}"</pre>

Option	Description	Example
cli-jsonInput	Specifies a JSON file to pass API parameters. If a cloud service API has too many parameters, you can define the parameters in a JSON file. KooCLI then parses the parameters in the file.	hcloud ECS CreateServers --cli-region="ap-southeast-1" --cli-read-timeout=60 --cli-jsonInput=C:\cli\Ecs_CreateServers.json
cli-connect-timeout and cli-read-timeout	Request timeouts. • cli-connect-timeout: request connection timeout, in seconds. The default value is 5, and the minimum value is 1. • cli-read-timeout: I/O timeout, in seconds. The default value is 10, and the minimum value is 1.	hcloud ECS DeleteServerPassword --cli-region="ap-southeast-1" --project_id="2cc60f5*****efa5019ef" --server_id="e6b99563-****-****-****-1820d4fd2a67" --cli-connect-timeout=10 --cli-read-timeout=15
cli-retry-count	Number of connection attempts. The system retries automatically if the connection times out. The value ranges from 0 to 5, and the default value is 0 .	hcloud RDS ListInstances --cli-region="ap-southeast-1" --Content-Type="application/json" --project_id="2cc60*****caefa5019ef" --cli-retry-count=3
cli-skip-secure-verify	Specifies whether to skip HTTPS certificate verification (not recommended). The value can be true or false . The default value is false . Skipping certificate verification has security risks. If you set this parameter to true , KooCLI will display a message asking for your confirmation.	hcloud ECS NovaListServers --cli-region="ap-southeast-1" --project_id="2cc6*****6caefa5019ef" --cli-skip-secure-verify=true

Option	Description	Example
cli-endpoint	Custom domain name. By default, requests are sent to the target cloud service in the relevant region. You can also specify an endpoint of this cloud service.	<code>hcloud IoTDA UpdateDevice --cli-region="ap-southeast-1" --description="test update device" --device_id="testz*****0802" --cli-endpoint="iot-mqttp.ap-southeast-1.myhuaweicloud.com"</code>
cli-waiter	Result polling. The parameters must be in JSON format and enclosed in double quotation marks.	<code>hcloud ECS NovaShowServer --cli-region="ap-southeast-1" --server_id="e6b99563-xxxx-xxxx-xxxx-1820d4fd2a6" --cli-waiter="{\"expr\": \"server.status\", \"to\": \"ACTIVE\", \"timeout\": 300}"</code>
cli-auth-type	AK/SK signing mode. Set this option only when you need to specify a special signing algorithm.	<code>hcloud IoTDA UpdateDevice --cli-region="ap-southeast-1" --description="test update device" --device_id="testz*****0802" --cli-endpoint="iot-mqttp.ap-southeast-1.myhuaweicloud.com" --cli-auth-type=derivedSign</code>
cli-x-project-id	Project ID, which is used for authentication.	<code>hcloud VPC ListAddressGroup/v3 --cli-region="ap-southeast-1" --cli-x-project-id="2cc60*****caefa5019ef"</code>

4.2 Printing Help Information

View help information of KooCLI commands.

For example, to view the help information about the RDS API for operation **ListCollations**, run this command:

```
hcloud RDS ListCollations --cli-region="ap-southeast-1" --help
```

4.3 Printing Debugging Information

Print debugging information during command execution.

To do so, add **--debug** in the command:

```
hcloud VPC ShowVpc/v3 --cli-region="ap-southeast-1" --project_id="0dd8cb*****19b5a84546" --vpc_id="0bbe****_****_****_****_235be6e7" --debug
[debug info] 2022/06/21 19:59:25 Read and connection timeouts are 40s and 30s respectively.
[debug info] 2022/06/21 19:59:25 URL: https://vpc.ap-southeast-1.myhuaweicloud.com/v3/0dd8cb*****19b5a84546/vpc/vpcs/0bbe****_****_****_****_235be6e7
[debug info] 2022/06/21 19:59:26 API response status code is 200.
[debug info] 2022/06/21 19:59:26 API response X-Request-Id is f9fd68*****2e48ec7f88.
{
  "vpc": {
    "id": "0bbe****_****_****_****_235be6e7",
    "name": "CCI-VPC-*****",
    "description": "",
    "cidr": "192.*.*.*/*",

```

```
"extend_cidrs": [],
"status": "ACTIVE",
"project_id": "0dd8cb*****19b5a84546",
"enterprise_project_id": "0",
"tags": [],
"created_at": "2022-05-10T02:53:42Z",
"updated_at": "2022-05-10T02:53:43Z",
"cloud_resources": [
  {
    "resource_type": "routetable",
    "resource_count": 1
  },
  {
    "resource_type": "virsubnet",
    "resource_count": 1
  }
],
"request_id": "f9fd68*****2e48ec7f88"
}
```

4.4 Generating API Input Parameter Skeleton in JSON Format

The **skeleton** option is used to generate an API input parameter skeleton in JSON format. Adding **--skeleton** to a command generates an API input parameter file in JSON format in the current directory. Specify parameter values, and pass them using **--cli-jsonInput=\${JSON file name}** to call the API.

```
hcloud RDS CreateConfiguration --cli-region="ap-southeast-1" --skeleton
Skeleton generated in JSON format. Path: C:\cli\RDS_CreateConfiguration_cn-20231108093800.json
```

The JSON file contains the following content:

```
{
  "header": {
    "X-Language": ""
  },
  "path": {
    "project_id": ""
  },
  "body": {
    "datastore": {
      "type": "",
      "version": ""
    },
    "description": "",
    "name": "",
    "values": {
      "": ""
    }
  }
}

----- After the skeleton is filled in, delete this line and the following parameter
description: -----
{
  "header": {
    "X-Language": {
      "Required": false,
      "ParamType": "string",
      "Usage": "Language",
      "EnumValue": [
        "zh-cn",
        "en-us"
      ]
    }
  }
}
```

```
    },
    "path": {
      "project_id": {
        "Required": true,
        "ParamType": "string",
        "Usage": "Project ID"
      }
    },
    "body": {
      "datastore": {
        "type": {
          "Required": true,
          "ParamType": "string",
          "Usage": "DB engine. Its value can be any of the following and is case-insensitive:\n- MySQL\n- PostgreSQL\n- SQLServer\n- MariaDB",
          "EnumValue": [
            "MySQL",
            "PostgreSQL",
            "SQLServer",
            "MariaDB"
          ]
        },
        "version": {
          "Required": true,
          "ParamType": "string",
          "Usage": "DB engine version\n\n - MySQL: 5.6 and 5.7 are supported. Example value: 5.7.
MySQL 8.0 is available only for users with the required permission. You can contact Huawei Cloud customer
service to apply for the permission.\n\n - PostgreSQL: 9.5, 9.6, 10, and 11 are supported. Example value:
9.6.\n\n - Microsoft SQL Server: 2017 Enterprise Edition, 2017 Standard Edition, 2017 Web Edition, 2014
Standard Edition, 2014 Enterprise Edition, 2016 Standard Edition, 2016 Enterprise Edition, 2012 Enterprise
Edition, 2012 Standard Edition, 2012 Web Edition, 2008 R2 Enterprise Edition, 2008 R2 Web Edition, 2014
Web Edition, and 2016 Web Edition are supported. Example value: 2014_SE.\n For example, enter 2017_SE
for 2017 Standard Edition, 2017_EE for 2017 Enterprise Edition, and 2017_WEB for 2017 Web Edition."
        }
      },
      "description": {
        "Required": false,
        "ParamType": "string",
        "Usage": "Parameter template description. It contains a maximum of 256 characters except the
following special characters: >!\<\"&'=. Its value is left blank by default."
      },
      "name": {
        "Required": true,
        "ParamType": "string",
        "Usage": "Parameter template name. It contains a maximum of 64 characters and can contain only
uppercase letters, lowercase letters, digits, hyphens (-), underscores (_), and periods (.)"
      },
      "values": {
        "": {
          "Required": false,
          "ParamType": "string",
          "Usage": "Parameter values defined by users based on the default parameter templates. If this
parameter is not specified, no parameter value is to be changed.\n\n - key: parameter name. For example,
in \"max_connections\": \"10\", the key is max_connections. If key is left blank, the parameter value is not to
be changed. If key is not empty, the parameter value cannot be empty, either.\n - value: parameter value.
For example, in \"max_connections\": \"10\", the value is 10."
        }
      }
    }
  }
}
```

NOTICE

- The generated JSON file consists of two parts, separated by a divider line. The upper part is the API parameter skeleton for **cli-jsonInput**, and the lower part shows parameter descriptions and value requirements. Specify the parameter values in the skeleton in the upper part by referring to the parameter description.
- After filling in the JSON file, delete the dividing line and the content in the lower part. Parameters with no value specified must be deleted.

For example, after you specify parameters in the preceding file, the file content is as follows:

```
{
  "path": {
    "project_id": "Odd8cb*****19b5a84546"
  },
  "body": {
    "datastore": {
      "type": "MySQL",
      "version": "5.7"
    },
    "description": "test create configuration",
    "name": "test-001",
    "values": {
      "max_connections": "10"
    }
  }
}
```

Put the file in the following command using **--cli-jsonInput** and run this command to call the API:

```
hcloud RDS CreateConfiguration --cli-region="ap-southeast-1" --cli-
jsonInput="RDS_CreateConfiguration_cn-20231108093800.json" --dryrun
API parameters transferred for the path, query, body, formData, header, and cookies in cli-jsonInput take
precedence over command transfers.
----- The execution is eliminated in dry-run mode. Current request: -----
POST https://rds.ap-southeast-1.myhuaweicloud.com/v3/Odd8cb*****19b5a84546/configurations
Content-Type: application/json
X-Project-Id: Odd8cb*****19b5a84546
X-Sdk-Date: 20220621T103331Z
Authorization: ****

{
  "datastore": {
    "type": "MySQL",
    "version": "5.7"
  },
  "description": "test create configuration",
  "name": "test-001",
  "values": {
    "max_connections": "10"
  }
}
```

4.5 Checking Command

The **dryrun** option is used to check the correctness of a command.

Add **--dryrun** in a command. The system prints the request message after verification. It does not call the target API.

```
hcloud RDS CreateConfiguration --cli-region="ap-southeast-1" --
project_id="0dd8cb*****19b5a84546" --datastore.type="MySQL" --datastore.version="5.7" --
values.max_connections="10" --name="test-001" --description="test create configuration" --dryrun
----- The execution is eliminated in dry-run mode. Current request: -----
POST https://rds.ap-southeast-1.myhuaweicloud.com/v3/0dd8cb*****19b5a84546/configurations
Content-Type: application/json
X-Project-Id: 0dd8cb*****19b5a84546
X-Sdk-Date: 20220621T103331Z
Authorization: ****

{
  "datastore": {
    "type": "MySQL",
    "version": "5.7"
  },
  "description": "test create configuration",
  "name": "test-001",
  "values": {
    "max_connections": "10"
  }
}
```

4.6 Specifying Region

In addition to reading region information from the profile, KooCLI can also read the value of **cli-region** you enter in a command, as shown below:

```
hcloud EVS DeleteVolume --cli-region="ap-southeast-1" --volume_id="aed9****_****_****_****0e3219cf" --
project_id="0dd8cb*****19b5a84546"
{
  "job_id": "70a5****_****_****_****441e862b"
}
```

NOTE

Projects vary according to regions. Therefore, you need to specify the project ID when specifying a region.

4.7 Calling APIs with AK/SK in Non-configuration Mode

Add the **--cli-access-key**, **--cli-secret-key**, and **--cli-security-token** options in a command to call a cloud service API with an AK/SK **in non-configuration mode**.

By default, if only **--cli-access-key** and **--cli-secret-key** are used in a command, the AK/SK are taken as permanent ones.

```
hcloud RDS ListApiVersion --cli-region="ap-southeast-1" --cli-access-key=***** --cli-secret-key=*****
{
  "versions": [
    {
      "id": "v3",
      "links": [],
      "status": "CURRENT",
      "updated": "2019-01-15T12:00:00Z"
    },
    {
      "id": "v1",
      "links": [],
      "status": "DEPRECATED",
      "updated": "2017-02-07T17:34:02Z"
    }
  ]
}
```

By default, if **--cli-access-key**, **--cli-secret-key**, and **--cli-security-token** are used in a command, the AK/SK are taken as temporary ones.

```
hcloud RDS ListApiVersion --cli-region="ap-southeast-1" --cli-access-key=***** --cli-secret-key=***** --cli-security-token=*****
{
  "versions": [
    {
      "id": "v3",
      "links": [],
      "status": "CURRENT",
      "updated": "2019-01-15T12:00:00Z"
    },
    {
      "id": "v1",
      "links": [],
      "status": "DEPRECATED",
      "updated": "2017-02-07T17:34:02Z"
    }
  ]
}
```

4.8 Calling APIs Using an AssumeRole in Non-configuration Mode

After a delegating party creates an agency to delegate another account to manage its resources, the delegated party can append **--cli-agency-domain-id/--cli-agency-domain-name**, **--cli-agency-name**, and **--cli-source-profile** to a command to call cloud service APIs using an AssumeRole in non-configuration mode, and manage the delegating party's resources.

```
hcloud VPC ListAddressGroup/v3 --cli-region="ap-southeast-1" --project_id="2cc60*****caefa5019ef" --cli-agency-domain-id=13534326*****5cf67b --cli-agency-name=***** --cli-source-profile=test
{
  "request_id": "29ec21*****6d6b4cdd82",
  "address_groups": [],
  "page_info": {
    "current_count": 0
  }
}
```

NOTE

Among the preceding parameters, use **--cli-agency-domain-id/--cli-agency-domain-name** and **--cli-agency-name** at the same time. Use **--cli-source-profile** to specify the profile that stores the authentication information of the delegated party. The value of **--cli-source-profile** cannot be the current profile.

4.9 Specifying Account ID

When calling APIs of global services in AK/SK mode, an account ID (**cli-domain-id**) is required.

During API calling, KooCLI automatically obtains the account ID based on the user authentication information. You can also add the **--cli-domain-id** option in the command, as shown below:

```
hcloud CDN ListDomains --cli-region="ap-southeast-1" --cli-domain-id="08e09a6e*****1bb800"
{
  "total": 0,
```

```
"domains": null
}
```

4.10 Specifying Profile

KooCLI supports multiple profiles, where you can save your common information (such as AK/SK pairs and regions).

Use such information by specifying the profile name with **--cli-profile**. The following is a command example:

```
hcloud EVS ListSnapshots --cli-profile=test
```

4.11 Specifying Authentication Mode

KooCLI allows you to set the authentication mode to **AKSK**, **SSO**, **AssumeRole**, or **ecsAgency** in a profile. **AKSK** is recommended. If the profile you use contains [parameters related to different authentication modes](#), use **--cli-mode** to specify the mode that will be used.

```
hcloud CCE ListNodes --cluster_id="f288****_****_****_****ac101534" --  
project_id="0dd8cb*****19b5a84546" --cli-profile=test --cli-mode=AKSK
```

NOTE

When configuring a profile, use **--cli-profile** to specify the profile name and add authentication parameters corresponding to **--cli-mode**.

- If **--cli-mode** is **AKSK**, set both **--cli-access-key** and **--cli-secret-key**.
- If **--cli-mode** is **ecsAgency**, set **--cli-mode=ecsAgency**.
- If **--cli-mode** is **SSO**, configure both **--cli-sso-start-url** and **--cli-sso-region**.
- If **--cli-mode** is **AssumeRole**, configure **--cli-agency-domain-id/--cli-agency-domain-name**, **--cli-agency-name**, and **--cli-source-profile**.

4.12 Specifying Output Format

Use the **--cli-query** option in a command to pass [a JMESPath expression](#) so that you can extract key information from the return result. Use **--cli-output** to specify the output format of the response data and use **--cli-output-num** to specify whether to print the row numbers during table output.

Output Order

The sequence of parameters in the output result varies according to the JMESPath expression specified by **--cli-query**. The output result queried by some expressions does not contain the attribute names (parameter keys) of the output data. If you use these expressions, understand the sequence of the output data to facilitate data processing. The following table describes the output sequence of different types of JMESPath expressions.

Table 4-2 Data output sequence of different JMESPath expressions

JMESPath Expression Type	Example JMESPath Expression	Attribute Names Contained in JSON or Table Output	Attribute Names Contained in TSV Output	Data Output Sequence	Example Output
Object-level expression	--cli-query="items[0]"	Yes	No	In alphabetical order of the attribute names of the object	Example 1
Attribute-level expression without changing the attribute names	--cli-query="items[0].items[0].spec.flavor,metadata.uid"	No	No	In the attribute name order specified in the JMESPath expression	Example 2
Attribute-level expression with the attribute names changed	--cli-query="items[0].{Flavor:spec.flavor,ClusterID:metadata.uid}"	Yes	No	In alphabetical order of the new attribute names	Example 3

The following examples show the sequence of data output in JSON format through different JMESPath expressions:

- Example 1

When an object is specified, KooCLI outputs the attribute values in alphabetical order of the attribute names of the object. In this example, the object **items[0]** is specified. The attributes of the object are sorted in alphabetical order as follows: **apiVersion**, **kind**, **metadata**, **spec**, **status**. The output result is as follows:

```
hcloud CCE ListClusters --cli-region="ap-southeast-1" --type="VirtualMachine" --  
project_id="0dd8cb*****19b5a84546" --cli-query="items[0]"  
{  
  "apiVersion": "v3",  
  "kind": "Cluster",  
  "metadata": {  
    "creationTimestamp": "2022-05-13 08:51:58.252509 +0000 UTC",  
    "labels": {  
      "FeatureGates": "elbv3",  
    },  
    "name": "github-****-****",  
    "uid": "f288****-****-****-****ac101534",  
    "updateTimestamp": "2022-05-13 09:10:06.395875 +0000 UTC"  
  },  
  "spec": {  
    "authentication": {  
      "authenticatingProxy": {},  
      "mode": "rbac"  
    }  
  }  
}
```

```
,
"az": "multi_az",
"billingMode": 0,
"category": "CCE",
"containerNetwork": {
  "cidr": "10.*.*/**",
  "mode": "vpc-router"
},
"eniNetwork": {},
"extendParam": {
  "alpha.cce/fixPoolMask": "25",
  "kubernetes.io/cpuManagerPolicy": "",
  "upgradefrom": ""
},
"flavor": "cce.s2.small",
"hostNetwork": {
  "SecurityGroup": "653e****_****_****_****6a23eb7e",
  "subnet": "d5df****_****_****_****4955c724",
  "vpc": "c865****_****_****_****efe7e8d8"
},
"kubeProxyMode": "iptables",
"kubernetesSvcIpRange": "10.*.*.*/**",
"masters": [
  {
    "availabilityZone": "ap-southeast-1b"
  },
  {
    "availabilityZone": "ap-southeast-1a"
  }
],
"supportIstio": true,
"type": "VirtualMachine",
"version": "v1.19.10-r0"
},
"status": {
  "endpoints": [
    {
      "type": "Internal",
      "url": "https://192.*.*.*:5443"
    },
    {
      "type": "External",
      "url": "https://121.*.*.*:5443"
    }
  ],
  "phase": "Available"
}
}
```

- Example 2

When specific attributes of an object are specified without changing the attribute names, KooCLI outputs the attribute values in the order that the attribute names are specified. In this example, the expression **items[0].spec.flavor,metadata.uid** specifies the child attribute **flavor** of **spec** and **uid** of **metadata** under the object **items[0]**. In the output result, the value of **spec.flavor** is displayed prior to that of **metadata.uid**.

```
hcloud CCE ListClusters --cli-region="ap-southeast-1" --type="VirtualMachine" --
project_id="Odd8cb*****19b5a84546" --cli-query="items[0].spec.flavor,metadata.uid"
[
  "cce.s2.small",
  "f288****_****_****_****ac101534"
]
```

- Example 3

When specific attributes of an object are specified with the attribute names changed, KooCLI outputs the attribute values in alphabetical order of the new

attribute names. In this example, the child attributes of two attributes under the object **items[0]** are specified. The expression **items[0]**.

{Flavor:spec.flavor,ClusterID:metadata.uid} renames the **flavor** attribute of **spec Flavor** and renames the **uid** attribute of **metadata ClusterID**. The new attributes are sorted in alphabetical order as follows: **ClusterID, Flavor**.

Therefore, the output result is as follows:

```
hcloud CCE ListClusters --cli-region="ap-southeast-1" --type="VirtualMachine" --
project_id="Odd8cb*****19b5a84546" --cli-query="items[0].
{Flavor:spec.flavor,ClusterID:metadata.uid}"
{
  "ClusterID": "f288****_****_****_****_****ac101534",
  "Flavor": "cce.s2.small"
}
```

Output Format

Use **--cli-output** to specify the output format. The value can be **json**, **table**, or **tsv**.

- When the value of **--cli-output** is **json**:

The result is output in JSON format as follows:

```
hcloud configure list --cli-output=json --cli-query="profiles[]".
{Name:name,Mode:mode,Ak:accessKeyId,SK:secretAccessKey}"
[
  {
    "Ak": "8NV****IOV",
    "Mode": "AKSK",
    "Name": "test",
    "SK": "*****"
  },
  {
    "Ak": "H9N****MXW",
    "Mode": "AKSK",
    "Name": "default",
    "SK": "*****"
  }
]
```

- When the value of **--cli-output** is **table**:

The result is output in a table as follows:

```
C:\cli>hcloud configure list --cli-output=table --cli-query="profiles[]". {Name:name, Mode:mode,
AK:accessKeyId, SK:secretAccessKey}"
+----+-----+-----+-----+-----+
| No. | AK    | Mode | Name  | SK    |
+----+-----+-----+-----+-----+
| 1   | H9N***MXW | AKSK | default | ***** |
+----+-----+-----+-----+-----+
| 2   | 8NV***IOV | AKSK | test   | ***** |
+----+-----+-----+-----+-----+
C:\cli>
```

In the meantime, you can use **--cli-output-num** to specify whether to print the row numbers.

```
C:\cli>hcloud configure list --cli-output=table --cli-query="profiles[]". {Name:name, Mode:mode,
AK:accessKeyId, SK:secretAccessKey}" --cli-output-num=false
+-----+-----+-----+-----+
| AK    | Mode | Name  | SK    |
+-----+-----+-----+-----+
| H9N***MXW | AKSK | default | ***** |
+-----+-----+-----+-----+
| 8NV***IOV | AKSK | test   | ***** |
+-----+-----+-----+-----+
C:\cli>
```

- When the value of **--cli-output** is **tsv**:

The result is output in TSV format as follows:

```
C:\cli>hcloud configure list --cli-output=tsv --cli-query="profiles[]. {Name:name, Mode:mode,
AK:accessKeyId, SK:secretAccessKey}"
H9N***MXW      AKSK      default  ****
8NV***IOV      AKSK      test     ****
C:\cli>
```

In TSV format output, data values are separated by tab characters and line breaks, and no extra symbols are included. You can use the output result for other commands. The TSV output result does not contain table headers. To prevent confusion when using the data for other commands, you need to understand the data output sequence of different types of JMESPath expressions. For details, see [Output Order](#).

If multiple attributes are specified but not renamed in a JMESPath expression for TSV format output, attributes enclosed in square brackets ([]) will be displayed in new lines. The following is an example:

```
C:\cli>hcloud configure list --cli-output=tsv --cli-query="profiles[]. [[name], mode, accessKeyId, secretAccessKey]"
AKSK      H9N***MXW      ****
default
AKSK      8NV***IOV      ****
test
C:\cli>
```

In this example, the attributes **name**, **mode**, **accessKeyId**, and **secretAccessKey** are specified. Attribute **name** is output in a new line. The attributes are output in the order that they appear in the command. The first line displays the values of **mode**, **accessKeyId**, and **secretAccessKey**, and the second line displays the value of **name**.

NOTE

For details about other precautions for using **--cli-query**, **--cli-output**, and **--cli-output-num**, see [FAQs](#).

4.13 Passing API Parameters with JSON File

When calling a cloud service API that has too many parameters, use **--cli-jsonInput** to input some or all parameters of the API through a JSON file rather than directly entering the parameters in the command. Enter other parameters, such as [KooCLI system parameters](#) and API parameters that are not included in the JSON file, in the command. Example:

```
hcloud ECS CreateServers --cli-region="ap-southeast-1" --cli-read-timeout=60 --cli-jsonInput=C:\cli
\Ecs_CreateServers.json
{
  "job_id": "ff808082*****ae0646",
  "serverIds": [
    "dd86****_****_****_****_****g1527651"
  ]
}
```

To pass API parameters through **--cli-jsonInput**, compile a JSON file and place each API parameter in the corresponding key based on its location in the request.

You can construct a JSON file using the following methods:

Using --skeleton

Use **--skeleton** to **generate and fill an API input parameter skeleton file in JSON format**, and pass the file using **--cli-jsonInput=\${JSON file name}**.

Using API Explorer

Obtain API parameters in API Explorer and write them into a JSON file.

Step 1 Add **--help** to the end of the original command and run the command. Then view the location of each API parameter in **Params** of the command output.

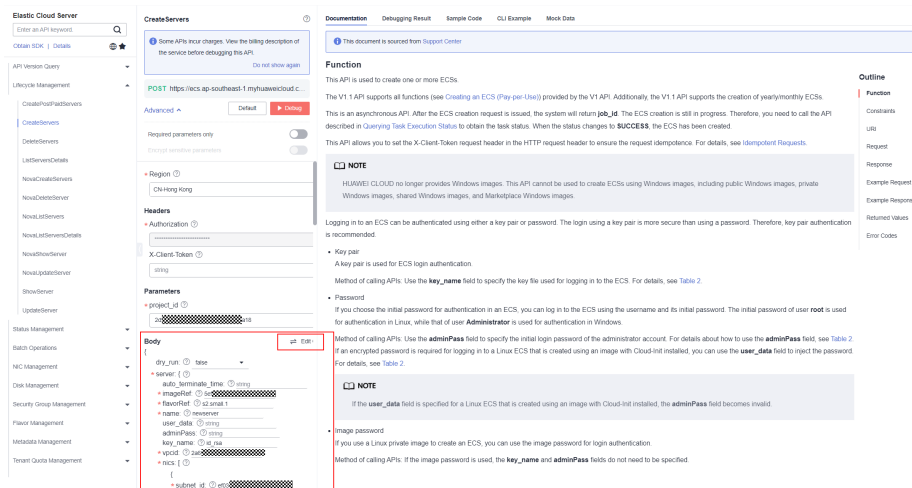
Step 2 Create a JSON file and name it in the format **\${Service name}_\${API name}.json**. Add the following content to the file:

```
{
  "header": {},
  "path": {},
  "query": {},
  "formData": {},
  "cookie": {},
  "body": {}
}
```

Step 3 Set parameters based on the key order in the JSON file.

- For **non-body** parameters, add **parameter-value** pairs in the relevant curly brackets ({}). Do not add any commas between the last pair and the right bracket ({}).
- For **body** parameters, set them in **API Explorer** and then click **Edit Code**, as shown in **the following figure**. Copy the JSON format parameters to the **body** key in the JSON file while **keeping only one pair of brackets**.

Figure 4-1 Obtaining body parameters for cli-jsonInput



Step 4 If there is no parameter next to a key, **delete the entire row where the key locates**. When deleting the last key, delete the comma (,) between the outermost and previous right brackets ({}).

Step 5 In your KooCLI command, use **--cli-jsonInput=\${JSON file location}** to pass API parameters, and run the command.

In the preceding example, the **Ecs_CreateServers.json** file passed by **--cli-jsonInput=C:\cli\Ecs_CreateServers.json** should be as follows:

```
{
  "path": {
    "project_id": "0dd8cb41*****a84546"
  },
  "body": {
    "server": {
      "adminPass": "wh*****",
      "auto_terminate_time": "2022-01-19T03:30:52Z",
      "availability_zone": "ap-southeast-1a",
      "data_volumes": [
        {
          "multiattach": true,
          "shareable": true,
          "size": 100,
          "volumetype": "SATA"
        }
      ],
      "flavorRef": "2d53****_****_****_****_****257bf163",
      "imageRef": "7059****_****_****_****_****0b5e9e4c",
      "name": "ecs_server_01",
      "nics": [
        {
          "ipv6_enable": true,
          "subnet_id": "4eb2****_****_****_****_****ff9a042d"
        }
      ],
      "publicip": {
        "eip": {
          "bandwidth": {
            "sharetype": "PER",
            "size": 30
          },
          "iptype": "5_sbgp"
        }
      },
      "root_volume": {
        "volumetype": "SATA"
      },
      "server_tags": [
        {
          "key": "date",
          "value": "211102"
        }
      ],
      "vpcid": "5aa5****_****_****_****_****1df05a3a"
    }
  }
}
```

----End

NOTICE

For more precautions, see [FAQs](#).

4.14 Specifying Request Timeouts

The **cli-connect-timeout** and **cli-read-timeout** options are used to set request timeouts. For the request connection timeout (**--cli-connect-timeout**), the default value is 5s. For the I/O timeout (**--cli-read-timeout**), the default value is 10s.

```
hcloud ECS DeleteServerPassword --cli-region="ap-southeast-1" --
project_id="2cc60f5*****efa5019ef" --server_id="e6b99563-****_****_****-1820d4fd2a67" --cli-connect-
timeout=10 --cli-read-timeout=15
```

You can use either **--cli-connect-timeout** or **--cli-read-timeout** or use both options in a command.

4.15 Specifying Retry Count

The **cli-retry-count** option specifies the number of connection attempts. The system retries automatically if a request times out due to a network connection problem. The value range of **--cli-retry-count** is 0 to 5, and the default value is 0.

- If a request fails due to abnormal network connection, the following information is displayed on KooCLI:

```
hcloud RDS ListInstances --cli-region="ap-southeast-1" --Content-Type="application/json" --
project_id="2cc60*****caefa5019ef" --cli-retry-count=3
[NETWORK_ERROR] Connection timed out 4 consecutive times (reconnection attempts: 3). Check
network connectivity.
```
- If the network connection is normal, the following information is displayed:

```
hcloud RDS ListInstances --cli-region="ap-southeast-1" --Content-Type="application/json" --
project_id="2cc60*****caefa5019ef" --cli-retry-count=3
{
  "instances": [],
  "total_count": 0
}
```



CAUTION

Setting the retry count with **--cli-retry-count** may cause idempotence and cyclic API calls. Exercise caution when using this option for resource creation APIs.

4.16 Skipping HTTPS Request Certificate Verification

The **cli-skip-secure-verify** option specifies whether to skip HTTPS certificate verification (**skipping is not recommended**). When you use KooCLI to call a cloud service API after [configuring the HTTP proxy](#), the error x509 may occur due to a certificate verification failure. To prevent this from happening, add **--cli-skip-secure-verify=true** to a command and then run the command. During the command execution, you will be prompted to confirm whether to skip HTTPS certificate verification.

```
hcloud ECS NovaListServers --cli-region="ap-southeast-1" --project_id="2cc6*****6caefa5019ef" --cli-
skip-secure-verify=true
Using '--cli-skip-secure-verify=true' will expose your private data to the public network and cause
interception risks. Continue? (y/N): y
{
  "servers": []
}
```

NOTE

However, using **--cli-skip-secure-verify=true** to skip HTTPS certificate verification will cause your private data to be exposed to the public network and a risk of theft. Therefore, you are **not advised** to use this option. To resolve this problem, **you are advised to** import the certificate issued by your company to the trusted CA certificate of the OS.

4.17 Specifying a Domain Name

Use **cli-endpoint** to specify a domain name of the target cloud service.

By default, KooCLI sends requests to the target cloud service in the relevant region. You can also specify an endpoint of this cloud service.

```
hcloud IoTDA UpdateDevice --cli-region="ap-southeast-1" --description="test update device" --
device_id="testz*****0802" --cli-endpoint="iot-mqts.ap-southeast-1.myhuaweicloud.com"
{
  "app_id": "103b*****6202",
  "app_name": "DefaultApp *****",
  "device_id": "testz*****0802",
  "node_id": "testz*****0802",
  "gateway_id": "testz*****0802",
  "device_name": "test*****0802",
  "node_type": "CATEWAY",
  "description": "test update device",
  "fw_version": null,
  "sw_version": null,
  "device_sdk_version": null,
  "auth_info": {
    "auth_type": "CERTIFICATES",
    "secret": null,
    "fingerprint": "e30db21*****b1772929c",
    "secure_access": true,
    "timeout": 0
  }
  "product_id": "50c1*****588e",
  "product_name": "50c1*****588e",
  "status": "INACTIVE",
  "create time": "20220801T085024z",
  "tags": [],
  "extension_info": {
    "sha1_fingerprint": "82cd23e2*****e089d59d",
    "sha256_fingerprint": "396ldb*****4a5f88ccb7"
  }
}
```

4.18 Polling Results

Use **cli-waiter** to poll results, and ensure that its parameters are in JSON format and enclosed in double quotation marks. These parameters help you poll results:

```
hcloud ECS NovaShowServer --cli-region="ap-southeast-1" --server_id="e6b99563-xxxx-xxxx-
xxxx-1820d4fd2a6" --cli-query="server.status" --cli-waiter="{\"expr\": \"server.status\", \"to\": \"ACTIVE
\", \"timeout\": \"300\"}"
Result polling ended. server.status is the expected ACTIVE.
ACTIVE
```

NOTE

In the value of **cli-waiter**, each field has a meaning as follows:

- **expr**: Required string, which is a JMESPath expression used to specify the field to poll for.
- **to**: Required string, which is the target value of the polled field.
- **timeout**: Optional int, which is the polling timeout in seconds. Range: 1 to 600. Default value: **180**.
- **interval**: Optional int, which is the polling interval in seconds. Range: 2 to 10. Default value: **5**.

 CAUTION

Polling results with **--cli-waiter** may cause idempotence and cyclic API calls. Use this option for query APIs.

4.19 Specify a Signing Algorithm

The **cli-auth-type** option specifies an AK/SK signing algorithm. Add this option when you need to specify a special signing algorithm.

The supported value is **derivedSign**. Use this value when [calling the IoTDA service with AK/SK](#).

```
hcloud IoTDA UpdateDevice --cli-region="ap-southeast-1" --description="test update device" --
device_id="testz*****0802" --cli-endpoint="iot-mqts.ap-southeast-1.myhuaweicloud.com" --cli-
auth-type=derivedSign
{
  "app_id": "103b*****6202",
  "app_name": "DefaultApp *****",
  "device_id": "testz*****0802",
  "node_id": "testz*****0802",
  "gateway_id": "testz*****0802",
  "device_name": "test*****0802",
  "node_type": "CATEWAY",
  "description": "test update device",
  "fw_version": null,
  "sw_version": null,
  "device_sdk_version": null,
  "auth_info": {
    "auth_type": "CERTIFICATES",
    "secret": null,
    "fingerprint": "e30db21*****b1772929c",
    "secure_access": true,
    "timeout": 0
  }
  "product_id": "50c1*****588e",
  "product_name": "50c1*****588e",
  "status": "INACTIVE",
  "create_time": "20220801T085024z",
  "tags": [],
  "extension_info": {
    "sha1_fingerprint": "82cd23e2*****e089d59d",
    "sha256_fingerprint": "396ldb*****4a5f88ccb7"
  }
}
```

4.20 Specifying a Project ID

cli-x-project-id specifies the project ID for authentication. This parameter is added when the project ID in the request header must be specified.

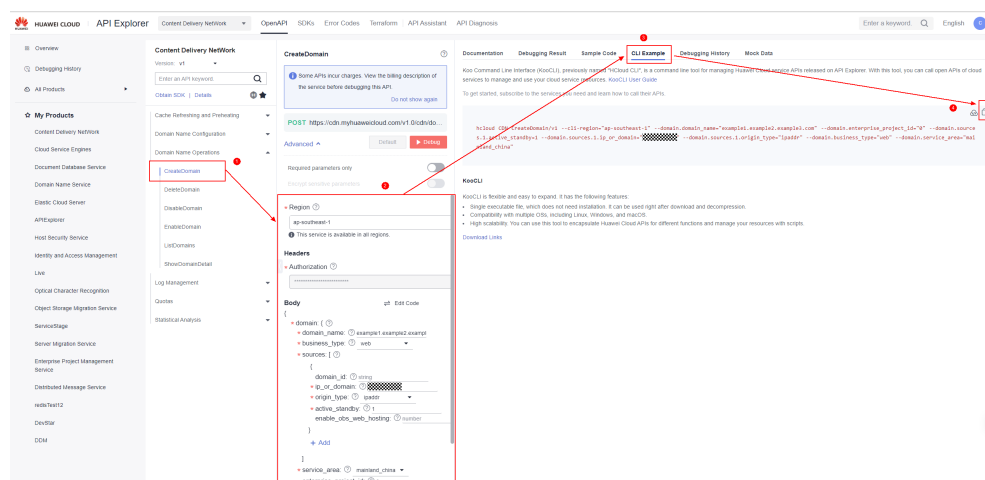
```
hcloud VPC ListAddressGroup/v3 --cli-region="ap-southeast-1" --cli-x-project-
id="2cc60*****caefa5019ef"
{
  "request_id": "29ec21*****6d6b4cdd82",
  "address_groups": [],
  "page_info": {
    "current_count": 0
  }
}
```

5 Obtaining CLI Examples on API Explorer

API Explorer supports online API debugging and generates KooCLI examples for cloud service APIs. After configuring the parameters of an API on API Explorer, you can copy and use the relevant **CLI Example**.

The following figure shows the CDN API for creating an acceleration domain name on API Explorer. After you set the request parameters, the CLI example will be updated accordingly. You can save the command with the set request parameters. Later, you can directly invoke this command in KooCLI or integrate it into a script.

Figure 5-1 Obtaining a CLI example on API Explorer



NOTE

- On the **API Explorer** console, only the parameters with values specified are displayed in the CLI example.
- A CLI example carries information such as a project ID and region. To use the example in another project or region, replace the project ID and region with the corresponding values.

6 Using KooCLI in Non-configuration Mode

6.1 Introduction

In KooCLI, you can call cloud service APIs using a profile or do so in non-configuration mode. When using KooCLI in non-configuration mode, you do not need to pass your authentication information through a profile. Instead, directly pass your authentication parameters in commands without adding any profiles. You can call cloud service APIs using any of the following authentication modes:

- [AK/SK Authentication](#)
- [ecsAgency Authentication](#)

When using KooCLI in non-configuration mode, note the [precautions](#) and understand the [priority of each authentication mode](#).

6.2 AK/SK Authentication

- Access key (permanent AK/SK)
Enter a permanent AK (**cli-access-key**) and SK (**cli-secret-key**) in a command to call a cloud service API.

```
hcloud ECS NovaListServers --cli-region="ap-southeast-1" --  
project_id="4ff018c3*****f31948" --cli-access-key=***** --cli-secret-key=*****
```
- Temporary security credentials (temporary AK/SK and SecurityToken)
Enter a temporary AK (**cli-access-key**), SK (**cli-secret-key**), and SecurityToken (**cli-security-token**) in a command to call a cloud service API.

```
hcloud ECS NovaListServers --cli-region="ap-southeast-1" --  
project_id="4ff018c3*****f31948" --cli-access-key=***** --cli-secret-key=***** --cli-security-  
token=*****
```

6.3 ecsAgency Authentication

Assume that you have created an ECS agency. When using KooCLI on an ECS, add **--cli-mode=ecsAgency** in a command so that KooCLI can use the ECS agency to automatically obtain a temporary AK/SK and SecurityToken for authentication.

To use this authentication mode, ensure that you have created an ECS agency. If no ECS agency is available, create one in IAM. For details, see [Cloud Service Delegation](#). Then add the agency in the **Management Information > Agency** area of the ECS details page.

7 Obtaining Authentication Information

7.1 Obtaining a Permanent AK/SK

Access keys (AK/SK) are an authentication mechanism of IAM. They are used to encrypt the signature of a request, ensuring that the request is secure and integral, and that identities of the request sender and receiver are correct.

- AK: a unique ID associated with an SK. It is used together with the SK to sign requests.
- SK: a key used together with the AK to sign requests. The AK and SK identify senders and prevent requests from being altered.

Constraint

You can create up to **two** access keys, which are permanently valid.

Viewing a Downloaded Access Key

If you have generated and downloaded an access key (AK/SK), find the local AK/SK file, which is generally named **credentials.csv**.

As shown in the following figure, the file contains a username, AK, and SK.

Figure 7-1 Content of the credential.csv file

	A	B	C
1	User Name	Access Key Id	Secret Access Key
2	AKIA	CI	zr17

Creating an Access Key

If no AK/SK file has been generated or can be found locally, create another access key.

Step 1 Log in to the console.

Step 2 On the top navigation menu, hover over the username and choose **My Credentials**.

Step 3 On the **My Credentials** page, choose **Access Keys > Create Access Key**.

Step 4 In the **Create Access Key** dialog box that is displayed, enter the password and verification code.

 NOTE

- If no email address and mobile number have been bound to your account, enter only the login password.
- If you have bound an email address or mobile number to your account, verify your identity using either of them

Step 5 Click **OK**.

Step 6 Save the access key as prompted. The access key is saved in the default download folder of the browser.

 NOTE

- Keep the access key secure. If you click **Cancel** in the download dialog box, the access keys will not be downloaded and cannot be downloaded later. You can create a new one if required.
- Rotate the access key (AK/SK) periodically.

Step 7 Open the downloaded **credentials.csv** file to obtain the access key (AK/SK).

----End

7.2 Obtaining an Account ID and Project ID

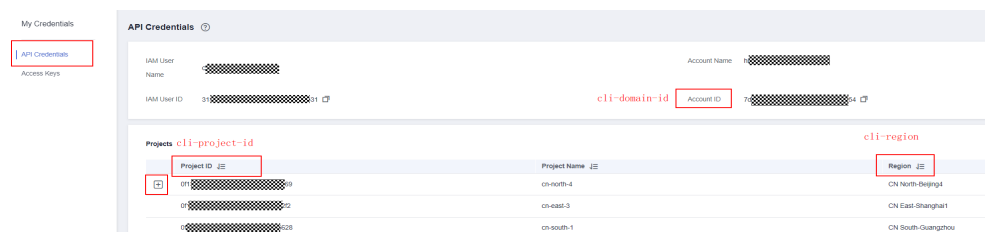
A project ID is required in most cases when you call a cloud service API. To obtain a project ID, perform the following operations:

Step 1 Log in to the management console.

Step 2 Click the username in the upper-right corner, choose **My Credentials** from the drop-down list, and view **Account ID (cli-domain-id)** and **Project ID (cli-project-id)** on the displayed page.

Projects physically isolate cloud server resources by region, and multiple projects can be created in the same region for more fine-grained isolation. As shown in the following figure, find the region where your server locates, obtain the corresponding project ID in the **Project ID** column, and click  on the left to obtain a subproject ID.

Figure 7-2 Viewing the account ID and project ID



 NOTE

During API calling, KooCLI automatically obtains the account ID and project ID from the request header based on the authentication information of the current user. Therefore, you do not need to enter them in a command.

----End

7.3 Obtaining a Region

You need to specify a region when calling cloud service APIs.

See [Regions and Endpoints](#).

 NOTE

KooCLI manages cloud service resources in a specified region.

7.4 Obtaining a Temporary AK/SK and SecurityToken

A temporary AK/SK and SecurityToken are issued by the system to IAM users and are valid for 15 minutes to 24 hours.

The temporary AK/SK and SecurityToken follow the principle of least privilege.

For details, see [Obtaining a Temporary Access Key and SecurityToken Through a Token](#).