Alibaba Cloud Elasticsearch

User Guide

Issue: 20190425

Legal disclaimer

Alibaba Cloud reminds you to carefully read and fully understand the terms and conditions of this legal disclaimer before you read or use this document. If you have read or used this document, it shall be deemed as your total acceptance of this legal disclaimer.

1. **You shall download and obtain this document from the Alibaba Cloud website or other Alibaba Cloud-authorized channels, and use this document for your own legal business activities only. The content of this document is considered confidential information of Alibaba Cloud. You shall strictly abide by the confidentiality obligations. No part of this document shall be disclosed or provided to any third party for use without the prior written consent of Alibaba Cloud.**
2. **No part of this document shall be excerpted, translated, reproduced, transmitted, or disseminated by any organization, company, or individual in any form or by any means without the prior written consent of Alibaba Cloud.**
3. **The content of this document may be changed due to product version upgrades , adjustments, or other reasons. Alibaba Cloud reserves the right to modify the content of this document without notice and the updated versions of this document will be occasionally released through Alibaba Cloud-authorized channels. You shall pay attention to the version changes of this document as they occur and download and obtain the most up-to-date version of this document from Alibaba Cloud-authorized channels.**
4. **This document serves only as a reference guide for your use of Alibaba Cloud products and services. Alibaba Cloud provides the document in the context that Alibaba Cloud products and services are provided on an "as is", "with all faults " and "as available" basis. Alibaba Cloud makes every effort to provide relevant operational guidance based on existing technologies. However, Alibaba Cloud hereby makes a clear statement that it in no way guarantees the accuracy, integrity , applicability, and reliability of the content of this document, either explicitly or implicitly. Alibaba Cloud shall not bear any liability for any errors or financial losses incurred by any organizations, companies, or individuals arising from their download, use, or trust in this document. Alibaba Cloud shall not, under any circumstances, bear responsibility for any indirect, consequential, exemplary, incidental, special, or punitive damages, including lost profits arising from the use**

or trust in this document, even if Alibaba Cloud has been notified of the possibility of such a loss.

5. By law, all the content of the Alibaba Cloud website, including but not limited to works, products, images, archives, information, materials, website architecture, website graphic layout, and webpage design, are intellectual property of Alibaba Cloud and/or its affiliates. This intellectual property includes, but is not limited to, trademark rights, patent rights, copyrights, and trade secrets. No part of the Alibaba Cloud website, product programs, or content shall be used, modified , reproduced, publicly transmitted, changed, disseminated, distributed, or published without the prior written consent of Alibaba Cloud and/or its affiliates . The names owned by Alibaba Cloud shall not be used, published, or reproduced for marketing, advertising, promotion, or other purposes without the prior written consent of Alibaba Cloud. The names owned by Alibaba Cloud include, but are not limited to, "Alibaba Cloud", "Aliyun", "HiChina", and other brands of Alibaba Cloud and/or its affiliates, which appear separately or in combination, as well as the auxiliary signs and patterns of the preceding brands, or anything similar to the company names, trade names, trademarks, product or service names, domain names, patterns, logos, marks, signs, or special descriptions that third parties identify as Alibaba Cloud and/or its affiliates).
6. Please contact Alibaba Cloud directly if you discover any errors in this document.

Generic conventions

Table -1: Style conventions

Style	Description	Example
	This warning information indicates a situation that will cause major system changes, faults, physical injuries, and other adverse results.	 Danger: Resetting will result in the loss of user configuration data.
	This warning information indicates a situation that may cause major system changes, faults, physical injuries, and other adverse results.	 Warning: Restarting will cause business interruption. About 10 minutes are required to restore business.
	This indicates warning information, supplementary instructions, and other content that the user must understand.	 Notice: Take the necessary precautions to save exported data containing sensitive information.
	This indicates supplemental instructions, best practices, tips, and other content that is good to know for the user.	 Note: You can use Ctrl + A to select all files.
>	Multi-level menu cascade.	Settings > Network > Set network type
Bold	It is used for buttons, menus , page names, and other UI elements.	Click OK.
`Courier font`	It is used for commands.	Run the `cd / d C :/ windows` command to enter the Windows system folder.
Italics	It is used for parameters and variables.	`bae log list --instanceid` *Instance_ID*
[] or [a|b]	It indicates that it is a optional value, and only one item can be selected.	`ipconfig` *[-all|-t]*

Style	Description	Example
{} or {a|b}	It indicates that it is a required value, and only one item can be selected.	`swich {stand | slave}`

Contents

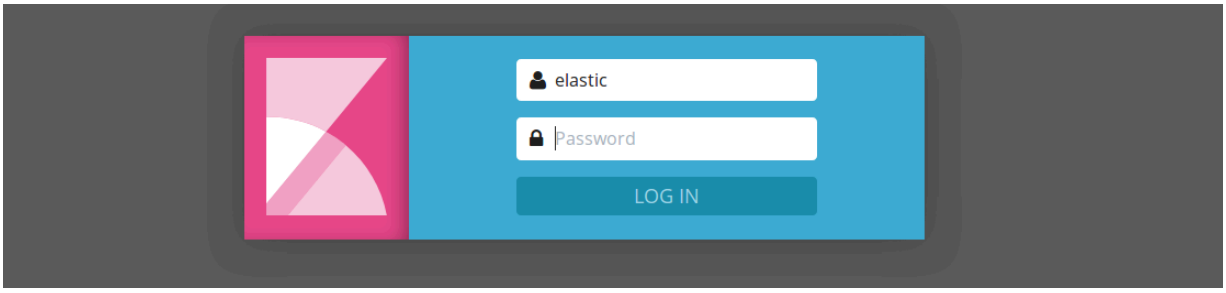
1 Instance management

1.1 Instance management

Elasticsearch instance management

Alibaba Cloud Elasticsearch supports multiple features for instance management, including Kibana console, instance monitoring, instance restart, refresh, and task list .

Kibana console



Elasticsearch provides the Kibana console for business scaling.

The Kibana console is a part of the Elasticsearch ecosystem, which has been seamlessly integrated into Elasticsearch. The Kibana console enables you to view the running status of your Elasticsearch instances and manage these instances.

Instance monitoring

Elasticsearch supports instance monitoring. You can customize alert thresholds and enable Elasticsearch to use SMS alerts when any exceptions have been detected. For more information, see *ES CloudMonitor alarm*.

Instance restart

This feature allows you to use the restart and force-restart method to restart an Elasticsearch instance. Select a restart method based on your business scenario.

Restart the agent

This method ensures service continuity by keeping at least one replica running on the Elasticsearch instance during the restart process. However, a restart using this method takes a long period of time.

 Note:

- Make sure that the health status of your Elasticsearch instance is green.
- The CPU and memory usage of the Elasticsearch instance will experience a usage spike during the restart process. This may affect the stability of your service for a short period of time.

Force-restart

This method may cause service instability on the Elasticsearch instance during the restart process. However, this method takes less time.

 Note:

When an Elasticsearch instance has a high disk usage, such as 85% or higher, the health status of the instance may change to yellow or red. You cannot restart an instance in red or yellow health status. To restart the instance, you must use the force-restart method.

- We recommend that you do not perform instance operations including node scaling, disk scaling, restart, password modification, and configuration modificati on when the health status of your Elasticsearch instance is yellow or red. Perform these operations when the health status of your instance is green.
- If you change the configuration of an unhealthy instance that contains two or more nodes, the instance will remain in the Applying status. To resolve this issue, submit a ticket.
- If you perform the update, restart, scaling, or password reset operation on an Elasticsearch instance that contains only one node, the service on the instance will become unavailable during the execution of the operation. To resolve this issue, create an Elasticsearch instance and migrate your service to the newly created instance.

Refresh

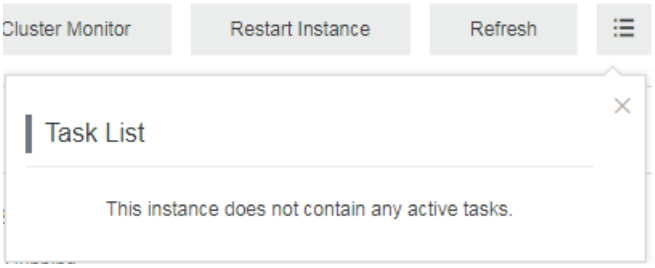
In certain cases, the console may fail to update the information. For example, the console may fail to update the status of an Alibaba Cloud Elasticsearch instance after

the instance has been successfully created. To resolve this issue, use the refresh function to manually refresh the status of the instance.

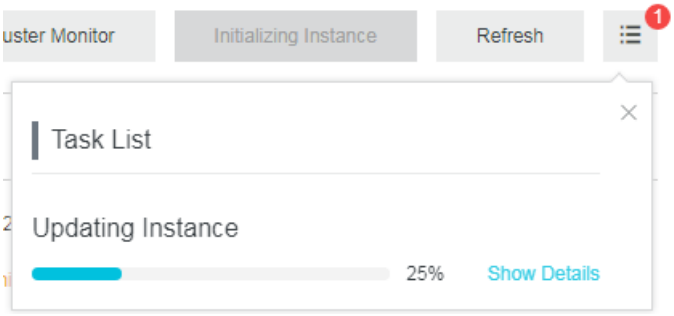
Task list

The Task list page shows running tasks on the current instance.

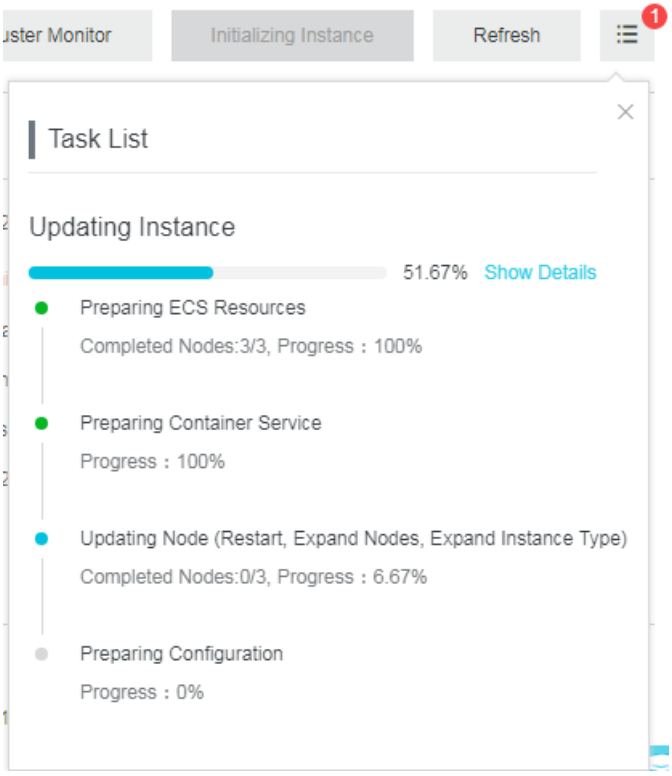
- No task is running on the current instance.



- Tasks running on the current instance.



- Show detailed information about a running task.



1.2 Basic information

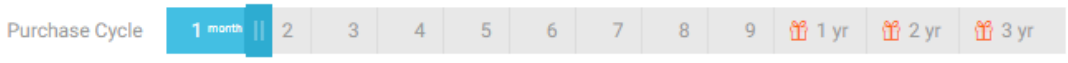
Prepayment (Subscription)

For basic information and parameters when using the Subscription billing method to purchase Alibaba Cloud Elasticsearch instances, see *Buy page parameters*.

Renewal

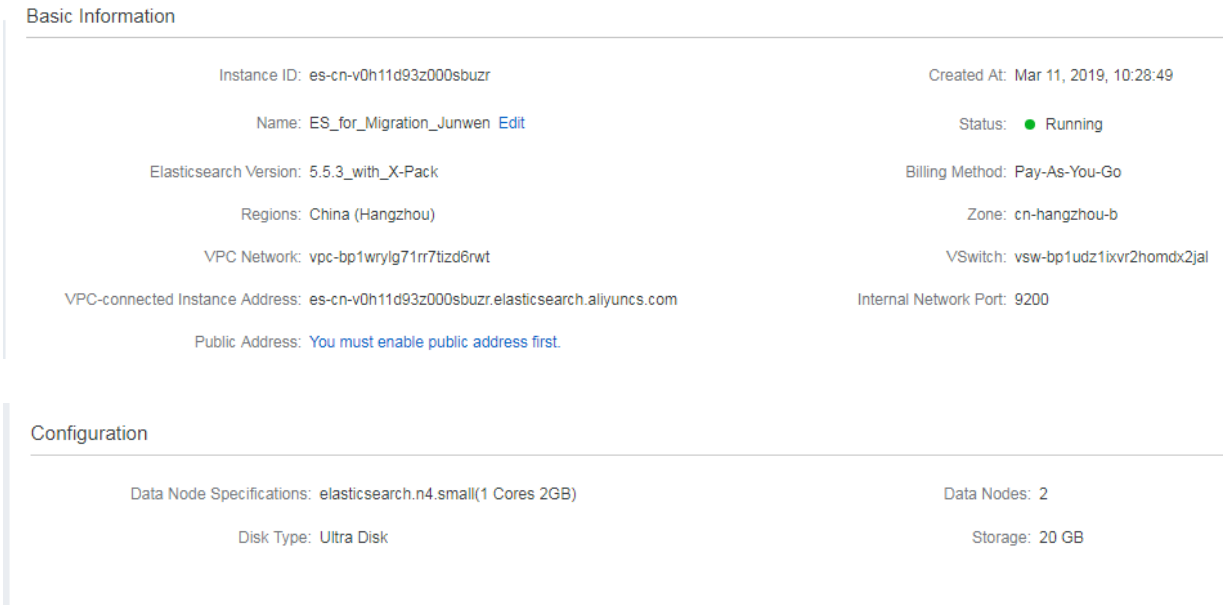
Drag the slider to the right to select the renewal duration. By default, the renewal duration is 1 month. Optional durations are 1 to 9 months or 1 to 3 years.

> **Note:**
>
> - Renewing an instance for 1 to 3 years will result in a discounted price.
> - The shortest renewal duration you can choose is 1 month.



Post-payment (Pay-As-You-Go)

For basic information and parameters when using the Pay-As-You-Go billing method to purchase Alibaba Cloud Elasticsearch instances, see *Buy page parameters*.



Switch from Pay-As-You-Go to Subscription

ES Pay-As-You-Go instances can be converted to Subscription instances. Click the Subscription Billing button. The process of changing billing methods is similar to that of renewal.

Cluster expansion

You can scale out by instance specification, node number, dedicated master node specification, or node storage space. For more information, see *Cluster upgrade*.

Name

By default, the name of an Elasticsearch instance is the same as its ID. Elasticsearch allows you to customize instance names and search for instances by name in the console.

Dedicated master node

You can select Dedicated Master Node on the Alibaba Cloud Elasticsearch purchase page and then purchase dedicated master nodes for your Elasticsearch instance. You can also purchase or scale out dedicated master nodes on the cluster expansion page. We recommend that you purchase dedicated master nodes to improve the stability of your service. For more information, see *Buy page parameters*.

Private address

You can use a private address to access an Elasticsearch instance from an ECS instance in a VPC-Connected network.

Private ports

Elasticsearch allows you to specify the following ports as private ports:

- Port `9200` (for HTTP).
- Port `9300` (for TCP), which is used to support X-Pack for ES 5.5.3.

Note:
The ES 6.3.2 with X-Pack version does not support the specification of port `9300`.

Notice:
Information security is not guaranteed when you access an Elasticsearch instance from the Internet. For secure access to Elasticsearch, we recommend that you purchase an ECS instance that meets the requirements of Alibaba Cloud Elasticsearch and use a private address to access Elasticsearch through a VPC.

Public address

You can use a public address to access an Elasticsearch instance from the Internet.

Public ports

Elasticsearch allows you to specify the following ports as public ports:

- Port `9200` (for HTTP).
- Port `9300` (for TCP), which mainly supports X-Pack for ES 5.5.3.

Note:

- The ES 6.3.2 with X-Pack version does not support the specification of port `9300`.
- To access an Elasticsearch instance from the Internet, you must first create a *Public IP address whitelist*. By default, no public addresses are allowed to access Elasticsearch.

Other parameters

For information about parameters not described in this document, see the parameter descriptions on the basic information page.

1.3 Cluster upgrade

Cluster upgrade includes upgrading of the instance specifications, number of nodes, dedicated master node specifications, number of client nodes, client node specifications, number of warm nodes, warm node specifications, warm node storage space, and storage space per node.

> Note:
> You may not be able to upgrade some of the cluster properties due to certain restrictions. For more information, see the following sections.

Current configuration

Click Cluster Extension to show the configuration of the current Elasticsearch instance.

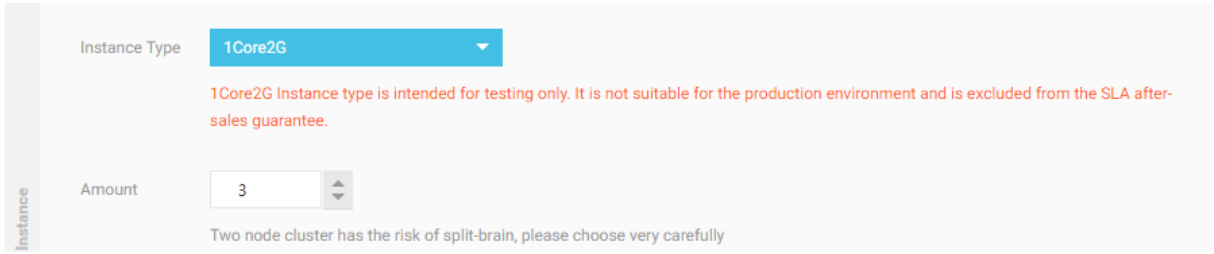
Change configuration

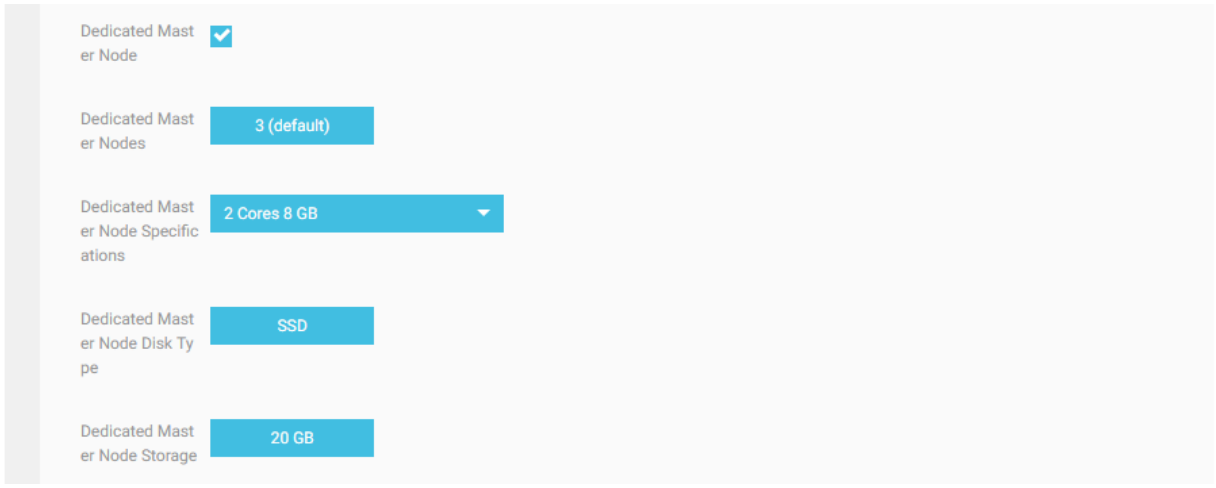
You can follow the tips on the Configuration Upgrade page to upgrade the configuration of a cluster based on your business needs. For more information about the parameters, see *Buy page parameters*.

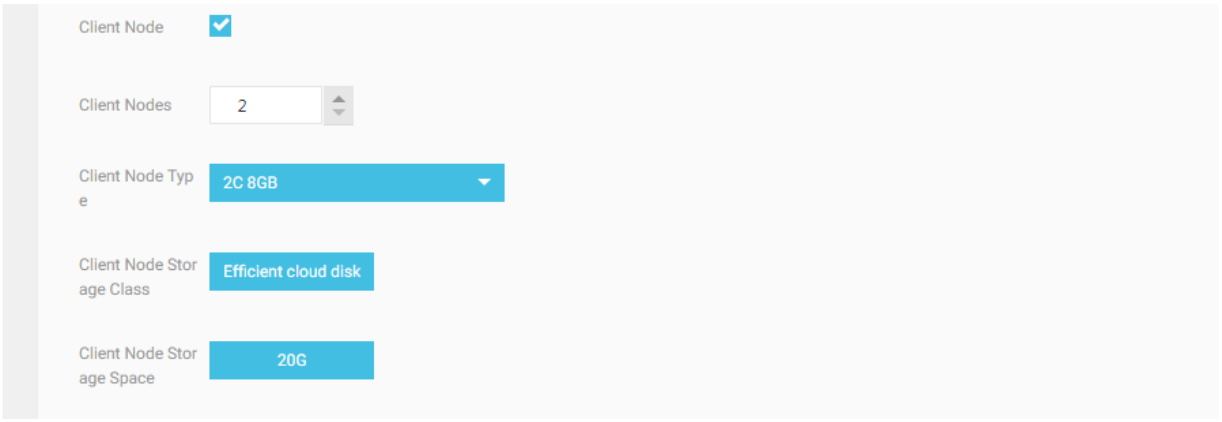
> Note:
> - For each upgrade, you can change only one of the cluster properties mentioned at the beginning of this topic.
> - You cannot change the storage type on the Configuration Upgrade page. You can only change the storage space.
> - The cluster upgrade operation restarts the corresponding Alibaba Cloud Elasticsearch instance.
> - Cluster downgrade, such as downgrading of the node count, storage space, and node specifications, is currently not supported.

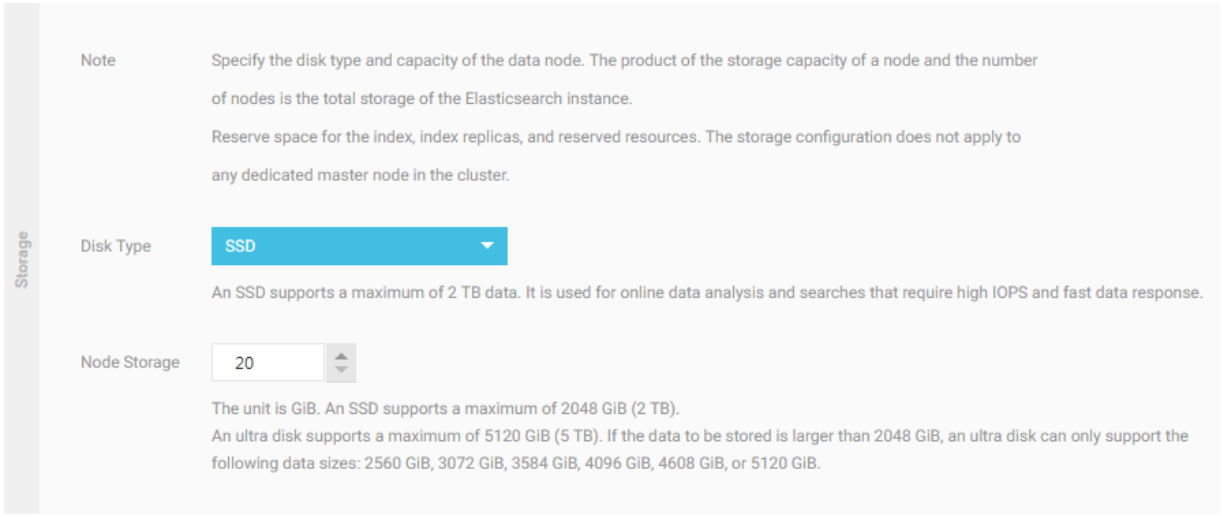
> Notice:

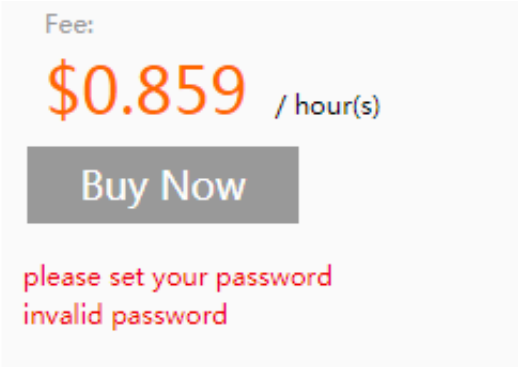
- If you have already purchased a dedicated master, changing the number of nodes does not restart the corresponding Alibaba Cloud Elasticsearch instance.
- To upgrade an Elasticsearch instance when the health status of the instance is not green, you must select Force Update. However, this may affect your business running on the Elasticsearch instance.
- If your business requires a cluster upgrade, we recommend that you make an upgrade assessment before upgrading the cluster.
- You can view the total cost of your cluster upgrade order on the Configuration Upgrade page in real time when changing the number of nodes.
- After you have submitted a cluster upgrade order, the upgrade Elasticsearch instance is billed based on the new configuration.











Instance types and specifications

You can follow the tips on the Configuration Upgrade page to change the specification of an Elasticsearch instance. For more information, see *Buy page parameters*.

Note:

- Data nodes that belong to the local disk specification family cannot be upgraded.
- You cannot modify the specification families.

Number of nodes

You can follow the tips on the Configuration Upgrade page to change the number of data nodes. For more information, see *Buy page parameters*.

Dedicated master nodes

You can select Dedicated Master Node on the Configuration Upgrade page to purchase dedicated master nodes or upgrade the specification of your purchased dedicated master nodes. The upgraded dedicated master nodes will be billed based on the new specification. For more information, see *Buy page parameters*.

 Note:

- If you have already purchased dedicated master nodes of 1-Core 2 GB, you can select Dedicated Master Node on the Configuration Upgrade page to repurchase dedicated master nodes with a higher specification. The dedicated master nodes will be billed based on the new specification. If you are using free dedicated master nodes, they will be billed after you upgrade their configuration.
- You can select Dedicated Master Node on the Configuration Upgrade page to upgrade the dedicated master node specification. The upgraded dedicated master nodes will be billed based on the new specification.
- You can select Dedicated Master Node on the Configuration Upgrade page to purchase dedicated master nodes or upgrade the specification of your purchased dedicated master nodes. By default, three dedicated master nodes of 2-Core 8 GB are used. The storage type of the dedicated master nodes is cloud disk. Each dedicated master node is assigned 20 GB of storage space.

Client nodes

You can select Client Node on the Configuration Upgrade page to purchase client nodes or upgrade the specification of your purchased client nodes. The upgraded client nodes will be billed based on the new specification. For more information, see *Buy page parameters*.

 Note:

Select Client Node on the Configuration Upgrade page to purchase client nodes or upgrade the specification of your purchased client nodes. By default, two client

nodes of 2-Core 8 GB are used. The storage type of the client nodes is cloud disk. Each client node is assigned 20 GB of storage space.

Warm nodes

You can select Warm Node on the Configuration Upgrade page to purchase warm nodes or upgrade the specification of your purchased warm nodes. The upgraded warm nodes will be billed based on the new specification. For more information, see *Buy page parameters*.

Note:
Select Warm Node on the Configuration Upgrade page to purchase client nodes or upgrade the specification of your purchased client nodes. By default, two warm nodes of 2-Core 8 GB are used. The storage type of the warm nodes is cloud disk. Each warm node is assigned 500 GB of storage space.

Restart

If the health status of your Elasticsearch instance is green, the Elasticsearch instance can provide services continuously during the upgrade restart process in most cases. You must make sure that your Elasticsearch instance has a minimum of one replica. The restart process may be time-consuming. Exceptions may occur during the restart process and the health status of your Elasticsearch instance may temporarily change to red.

Note:

- The nodes of an Elasticsearch instance may have a CPU and memory usage spike during the restart process. Your queries or pushing services may become unstable or fail. Typically, these services will recover after a short period of time. Exceptions may occur during the restart process and the health status of your Elasticsearch instance may temporarily change to red.
- You must make sure that the health status of your Elasticsearch instance is green.

Force update

If the health status of your Elasticsearch instance is red or yellow, this indicates that your services running on the instance have been severely affected. To resolve this issue, you must immediately upgrade your instance. You can select Force Update to

ignore the status of the Elasticsearch instance and forcibly upgrade the instance. The upgrade process takes only a short period of time.

Note:

- The Force Update operation will restart the Alibaba Cloud Elasticsearch instance.
- If you do not select Force Update, Elasticsearch uses the restart method to upgrade the instance.
- If the health status of your Elasticsearch instance is red or yellow, the Force Update option is automatically selected. You cannot use the restart method to upgrade the instance.
- The force update operation will make your services running on the Elasticsearch instance become unstable during the restart process.

Storage

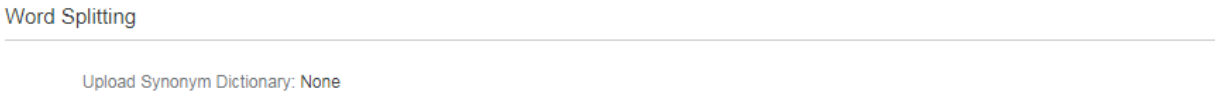
You can follow the tips on the Configuration Upgrade page to change the storage space of a node. For more information, see *Buy page parameters*.

Note:
You cannot change the storage space for a data node that is configured with an ultra disk larger than 2,048 GB.

1.4 Elasticsearch cluster configuration

Word splitting

This feature uses the synonym dictionary. New indexes will use the updated synonym dictionary. For more information, see *Configure synonyms for Elasticsearch instances*.



Note:

- After you upload and submit a synonym dictionary file, the Alibaba Cloud Elasticsearch instance will not restart immediately. It takes some time for the new configuration to take effect.

- If an index that is created before the uploaded synonym dictionary file takes effect needs to use synonyms, you must recreate the indexes and configure synonyms.

Write one synonym expression in each row and save the code as a `UTF - 8` encoded `. txt` file. Examples:

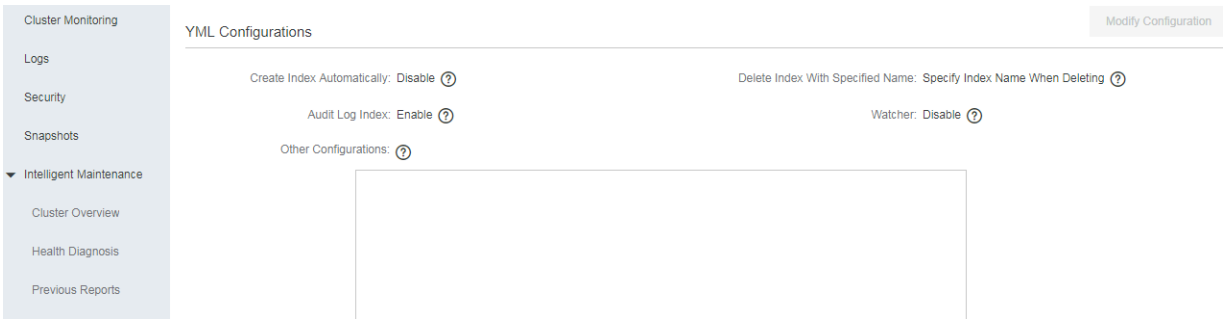
```
corn , maize => maize , corn
begin , start => start , begin
```

Configuration procedure:

1. Upload and save a synonym dictionary file in the Alibaba Cloud Elasticsearch console. Make sure that the uploaded file takes effect.
2. When you create an index and configure the `settings`, you need to specify the `" synonyms_p ath ": " analysis / your_dict_ name . txt "` path. Add a `mapping` for this index to configure synonyms for the specified field.
3. Confirm the synonyms and upload a file for testing.

YML configurations

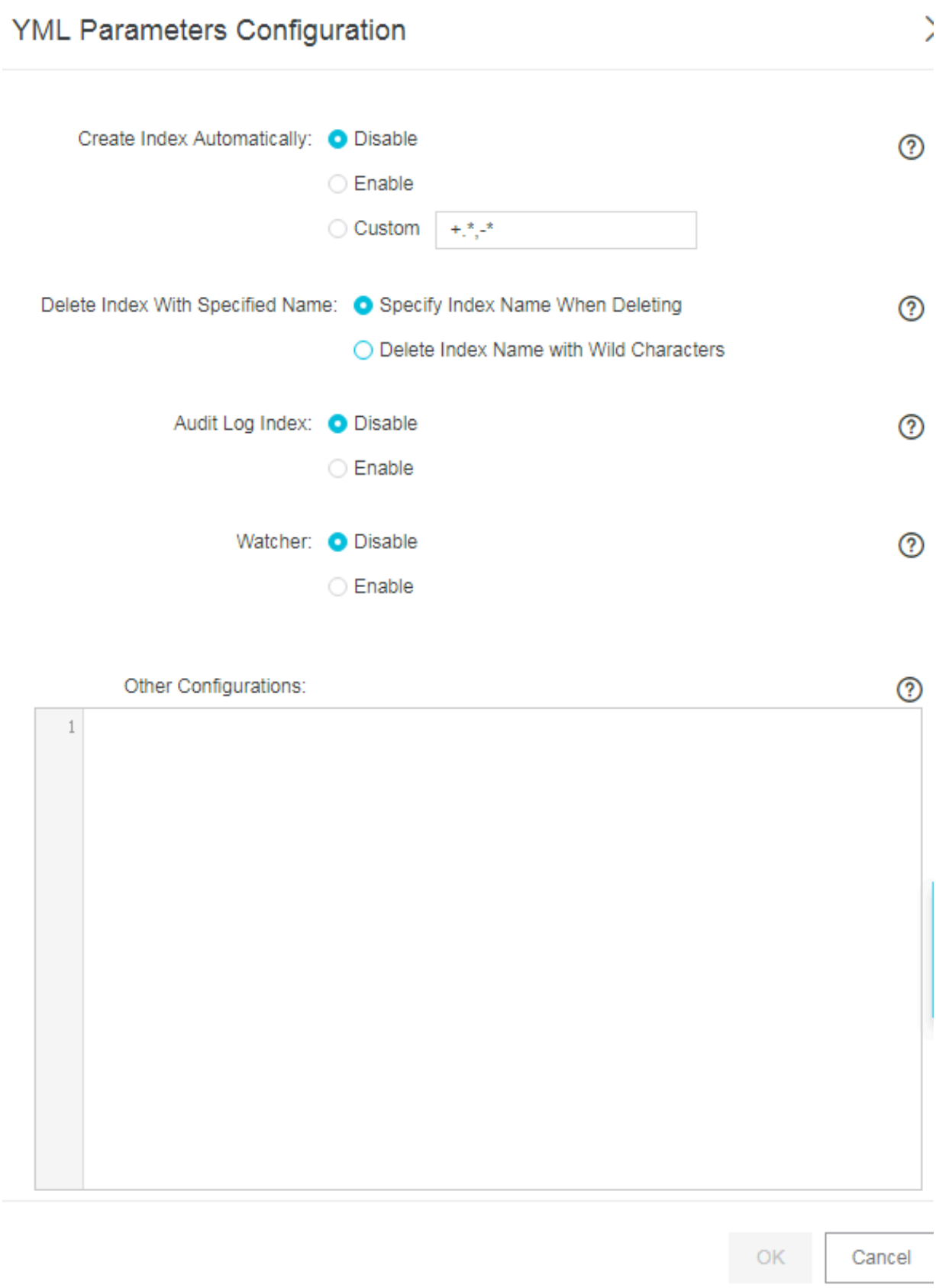
The YML Configurations page displays the settings of the current Alibaba Cloud Elasticsearch instance.



Modify YML configurations

After you modify the YML Configurations, you must restart the Alibaba Cloud Elasticsearch instance for the new configuration to take effect.

Note:
After you modify the YML Configurations, select This operation requires a restart of the instance. Exercise with caution. at the bottom of the page and click OK. The Alibaba Cloud Elasticsearch instance automatically restarts.



- **Create Index Automatically: if you enable this feature, it allows the system to automatically create new indexes if a new file is uploaded to the Alibaba Cloud Elasticsearch instance and no indexes have been created on the file. We**

recommend that you disable this feature. Indexes created by this feature may not meet your requirements.

- Delete Index With Specified Name: this feature indicates whether you are required to specify the name of the index that you need to delete. If you select Delete Index Name with Wild Characters, you can delete multiple indexes by using a wildcard character. Indexes that are deleted cannot be restored. Proceed with caution.
- Audit Log Index: if you enable this feature, index logs are created and stored when you create, delete, modify, or view an Alibaba Cloud Elasticsearch instance. These logs consume disk space and affect the performance. We recommend that you disable this feature. Proceed with caution.
- Watcher: if you enable this feature, it allows you to use the X-Pack Watcher feature. Make sure that you regularly clear the `. watcher - history *` index. This index consumes large amounts of disk space.
- Other Configurations: the following parameters are supported. For more information, see *YML configuration*.

   Note:
  Excluding the parameters that have an Alibaba Cloud Elasticsearch version specified, the remaining parameters can only be applied to Elasticsearch V5.5.3 and V6.3.2.

  - http.cors.enabled
  - http.cors.allow-origin
  - http.cors.max-age
  - http.cors.allow-methods
  - http.cors.allow-headers
  - http.cors.allow-credentials
  - reindex.remote.whitelist
  - action.auto_create_index
  - action.destructive_requires_name
  - thread_pool.bulk.queue_size (Elasticsearch V5.5.3 with X-Pack)
  - thread_pool.write.queue_size (Elasticsearch V6.3.2 with X-Pack)
  - thread_pool.search.queue_size

1.5 YML configuration

Customize CORS requests

For more configurations, visit the Elasticsearch official website and view the *HTTP* information.

Configuration information

- Configurations in the table below are custom HTTP-based configurations provided by Alibaba Cloud Elasticsearch.
- For the following configurations, only static configuration is supported. Dynamic configuration is not supported. Note that for the following configurations to take effect, you must add the configurations to the `elasticsea rch . yml` file.
- Cluster network settings are used for the following configurations. (*Network settings*)

Configuration item	Description
`http.cors.enabled`	A CORS (Cross-Origin Resource Sharing) configuration item, which can be used to enable or disable CORS resource accesses. In other words, this setting is used to determine whether to allow Elasticsearch to receive requests sent by browsers to access resources in different domains. If the parameter is set to `true`, Elasticsearch can process `OPTIONS` CORS requests. If the domain information in the sent request is already declared in `http.cors.allow-origin`, Elasticsearch adds `Access-Control-Allow-Origin` in the header to respond to the CORS request. If the parameter is set to `false` (which is the default value), Elasticsearch ignores the domain information in the request header, not adding the `Access-Control-Allow-Origin` to the header, disabling CORS access. If the client neither supports `pre-flight` requests that add the domain information header, nor checks `Access-Control-Allow-Origin` in the header of the packet returned from the server, then the secured CORS access will be affected. If Elasticsearch disables CORS access, then the client can only check whether a response is returned by sending the `OPTIONS` request.

Configuration item	Description
`http.cors.allow-origin`	A CORS resource configuration item, which can be used to specify requests from which domains are accepted. The parameter is left blank, by default, with no domain is allowed. If `/` is added before the parameter value, then the configuration is identified as a regular expression, which means that `HTTP` and `HTTPS` domain requests that follow the regular expression are supported. For example`/Https?:\/\/localhost(:[0-9]+)?/` means requests follow the regular expression can be responded to. `*` means that a configuration is valid and can be identified as enabling the cluster to support CORS requests from any domain, resulting in security risks to the Elasticsearch cluster.
`http.cors.max-age`	The browser can send an `OPTIONS` request to get the CORS configuration. `max-age` can be used to set how long the browser can retain the output result cache. The default value is `1728000` seconds (20 days).
`http.cors.allow-methods`	A request method configuration item. The optional values are `OPTIONS`, `HEAD`, `GET`, `POST`, `PUT`, and `DELETE`.
`http.cors.allow-headers`	A request header configuration item. The default value is `X-Requested-With, Content-Type, Content-Length`.
`http.cors.allow-credentials`	A credential configuration item, which is used to specify whether to return `Access-Control-Allow-Credentials` in the response header. If the parameter is set to `true`, Access-Control-Allow-Credentials is returned. The default value is `false`.

An example of custom cross-origin access configuration is as follows:

```
http . cors . enabled :  true
http . cors . allow - origin : "*"
http . cors . allow - headers : " X - Requested - With ,  Content -
Type ,  Content - Length ,  Authorizat  ion "
```

Customize remote re-indexing (whitelist)

The re-indexing component allows you to reconstruct the data index on the target remote Elasticsearch cluster. This function can work for all of the remote Elasticsea rch versions available, allowing you to index the data of earlier versions to the current version.

```
 POST   _   reindex
{
  " source ": {
    " remote ": {
      " Host ": " http : //  otherhost :  9200  ",
      " username ": " username ",
      " password ": " password ",
    },
    " index ": " source ",
    " query ": {
      " match ": {
        " test ": " data "
      }
    }
  },
  " dest ": {
    " index ": " test - 1 ",
  }
}
```

- `host` must contain the protocol supported, domain name, port, for example, `Https : //  otherhost :  9200` .
- `username` and `password` are optional. If the remote Elasticsearch server requires Basic Authorization, enter the username and password in the request. When use `Basic  Authorizat  ion` , also use the `https` protocol, otherwise the password will be transmitted as a text.
- The remote host address must be declared in `elasticsea  rch . yaml` by using the `reindex . remote . whitelist` attribute for the API to be called remotely. The combination of host and port is allowed. The combination of `host` and `port` is allowed. However, note that multiple host configurations must be separated by commas (,), for example,

  ```
  otherhost :  9200 ,  another :  9200 , 127 . 0  . 10 . **:  9200 ,
  ```

```
            localhost :**
```

). The whitelist does not identify the protocol and only uses the host and port information for the security policy configuration.

- If the host address is already listed in the whitelist, the `query` request will not be verified or modified. Rather, the request will be directly sent to the remote server.

 Note:

- Indexing data from a remote cluster is not supportedManual SlicingOrAutomatic Slicing. For more information, see *Manual slicing* or *Automatic slicing*.

Multiple indexes settings

The remote service uses a stack to cache indexed data. The default maximum size is `100 MB`. If the remote index contains a large document, set the size of batch settings to a small value.

In the example below, the size of multiple index settings is 10, which is the minimum value:

```
 POST   _   reindex
{
  " source ": {
    " remote ": {
      " host ": " http :// otherhost : 9200 "
    },
    " index ": " source ",
    " size ":  10 ,
    " query ": {
      " match ": {
        " test ": " data "
      }
    }
  },
  " dest ": {
    " index ": " test - 1 ",
  }
}
```

Timeout period

- Use `socket_tim eout` to set the read timeout period of `socket`. The default value is `30s`.
- Use `connect_ti meout` to set the connection timeout period. The default value is `1s`.

In the example below, the read timeout period of `socket` is one minute, and the connection timeout period is 10 seconds.

```
 POST   _   reindex
{
  " source ": {
    " remote ": {
      " host ": " http :// otherhost : 9200 ",
      " socket_tim  eout ": " 1m ",
      " connect_ti  meout ": " 10s "
    },
    " index ": " source ",
    " query ": {
      " match ": {
        " test ": " data "
      }
    }
  },
  " dest ": {
    " index ": " test - 1 ",
  }
}
```

Customize the access log

Enable auditing

The index auditing configuration is as follows.

```
xpack . security . audit . index . bulk_size :  5000
xpack . security . audit . index . events . emit_reque  st_body :
false
xpack . security . audit . index . events . exclude :  run_as_den
ied , anonymous_  access_den  ied , realm_auth  entication  _failed ,
access_den  ied , connection  _denied
xpack . security . audit . index . events . include :  authentica
tion_faile  d , access_gra  nted , tampered_r  equest , connection
_granted , run_as_gra  nted
xpack . security . audit . index . flush_inte  rval :  180s
xpack . security . audit . index . rollover :  hourly
xpack . security . audit . index . settings . index . number_of_
replicas :  1
xpack . security . audit . index . settings . index . number_of_
shards :  10
```

Index auditing output

Alibaba Cloud Elasticsearch instances do not support displaying request-related log files. Therefore, to view information about the Elasticsearch instance requests, such as the access_log, you must log in to the Elasticsearch console and enable the access log index function.

After this function is enabled, the access log is output to indexes on the Elasticsearch instance. The name of indexes starts with `. security_a  udit_log -*`.

Audit log indexing configuration

Note:

- Filtering is not supported during audits because sensitive data may be audited in plain text when the `request body` is included in audit events.
- Audit log indexing occupies Alibaba Cloud Elasticsearch instance storage space. You must manually clear old audit log indexes because no policy is available for clearing expired indexes.

Feature	Default value	[DO NOT TRANSLATE]
`xpack.security.audit.index.bulk_size`	`1,000`	Indicates how many audit events are batched into a single write file.
`xpack.security.audit.index.flush_interval`	`1s`	Indicates how often buffered events are flushed to the index.
`xpack.security.audit.index.rollover`	`daily`	Indicates how often to roll over to a new index. Options include `hourly`, `daily`, `weekly`, or `monthly`.
`Xpack.security.audit.index.events.include`	`anonymous_access_denied`, `authentication_failed`, `realm_authentication_failed`, `access_granted`, `access_denied`, `tampered_request`, `connection_granted`, `connection_denied`, `run_as_granted`, `run_as_denied`	Specifies the audit events to be indexed. For more information about audit event types, see *Audit event types*.

Feature	Default value	[DO NOT TRANSLATE]
`xpack.security.audit.index.events.exclude`		Excludes the specified auditing events from indexing.
`xpack.security.audit.index.events.emit_request_body`	`false`	Indicates whether to include the request body in REST requests in certain event types, such as `authentication_failed`.

Audit indexing settings

The configuration item `xpack.security.audit.index.settings` in the `elasticsearch.yml` file specifies the settings for the indexes in which the events are stored.

The following example sets both the number of shards and the number of replicas to `1` for the audit indexes.

```
xpack.security.audit.index.settings:
  index:
    number_of_shards: 1
    number_of_replicas: 1
```

Note:
You can pass custom settings to xpack.security.audit.index.settings when enabling audit indexing. Once you apply the change to the Elasticsearch instance, audit indexes will be available on the Elasticsearch instance. Otherwise, the elasticsearch instance audit log is set to the default `Number_of_shards: 5`, and `Number_of_replicas: 1`.

Remote audit log indexing settings

Indexing settings for remote audit logs are currently unavailable.

Customize thread pool queue size

You can set `Thread_pool.bulk.queue_size`, `Thread_pool.write.queue_size`, and `Thread_pool.search.queue_size` to customize the queue size of the write and search thread pools, respectively..

In the following example, both the write and search queue size are set to `500`.

Note:

The following parameters are not specifically identified for an ES version and by default are compatible with ES version 5.5.3 and 6.3.2.

```
thread_poo l . bulk . queue_size : 500 ( Only applicable to
the Elasticsea rch 5 . 5 . 3 with X - Pack version )
thread_poo l . write . queue_size : 500 ( Only applicable to
the Elasticsea rch 6 . 3 . 2 with X - Pack version )
thread_poo l . search . queue_size : 500
```

Parameter optimization

Configuration Item	Description
Index. codec	The ES data compression algorithm defaults to LZ4. Usually, by setting LZ4 to best_compression in a warm or cold cluster using a high-speed cloud disk, a higher compression ratio DEFLATE algorithm can be used. After the algorithm is changed, segment merges will use the newest version of the algorithm. Note that using best_compr ession will result in reduced write performance.

REST API settings

You can set the `index . codec` parameter by using REST API.

Note:

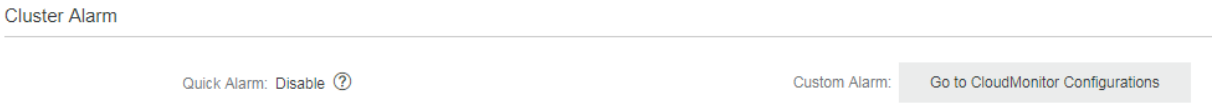
- `close` the corresponding index before running the command.
- $index_name: Replace with the index name you need to set.

```
PUT $ index_name / _ settings
{
  " index ": {
    " codec ": " best_compr ession "
  }
```

```
}
```

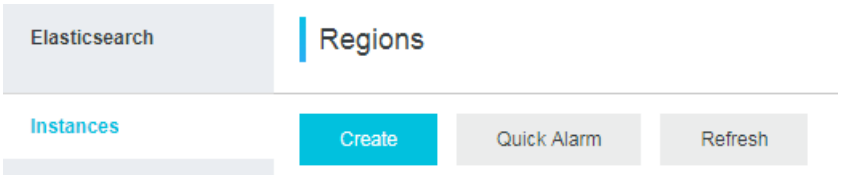
1.6 Cluster monitoring

Cluster alarm

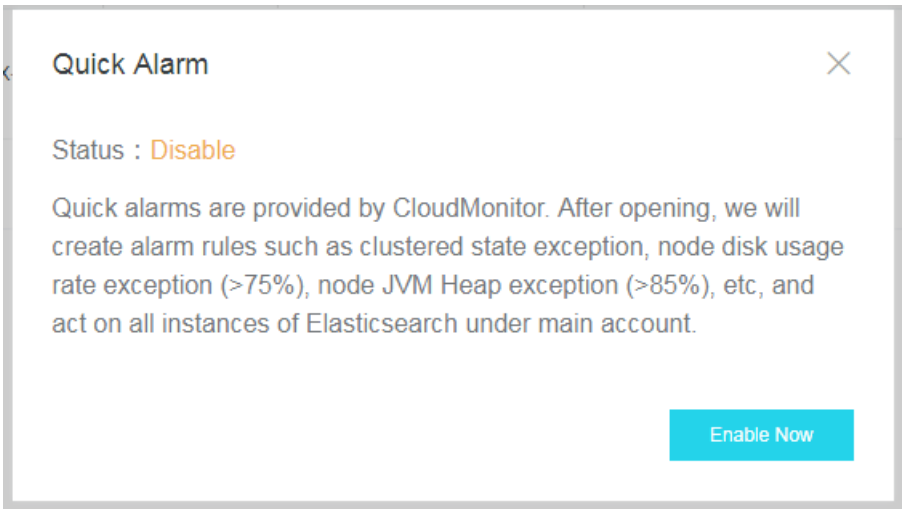


Quick alarm

1. Elasticsearch supports quick alarm. This feature is disabled by default. You can go to the clusters list page and click Quick Alarm to enable or disable this feature.



2. If this feature is disabled, click Quick Alarm, and then click Enable Now in the dialog box to manually enable it.



Custom alarms

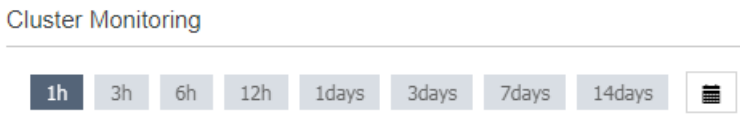
You can click Cluster Monitor to create custom alarm rules. For more information about creating alarm rules, see *ES CloudMonitor alarm*.

Cluster monitor

You can view Elasticsearch instance parameters and workloads.

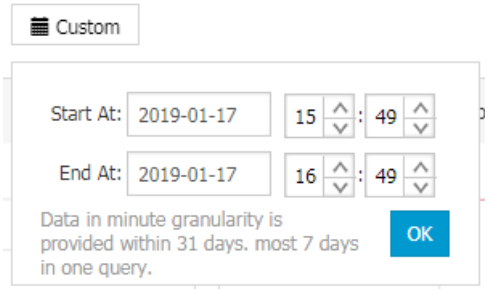
Preset time

You can click a time option to view cluster metrics that are collected in the specified time period.



Custom cluster monitoring time

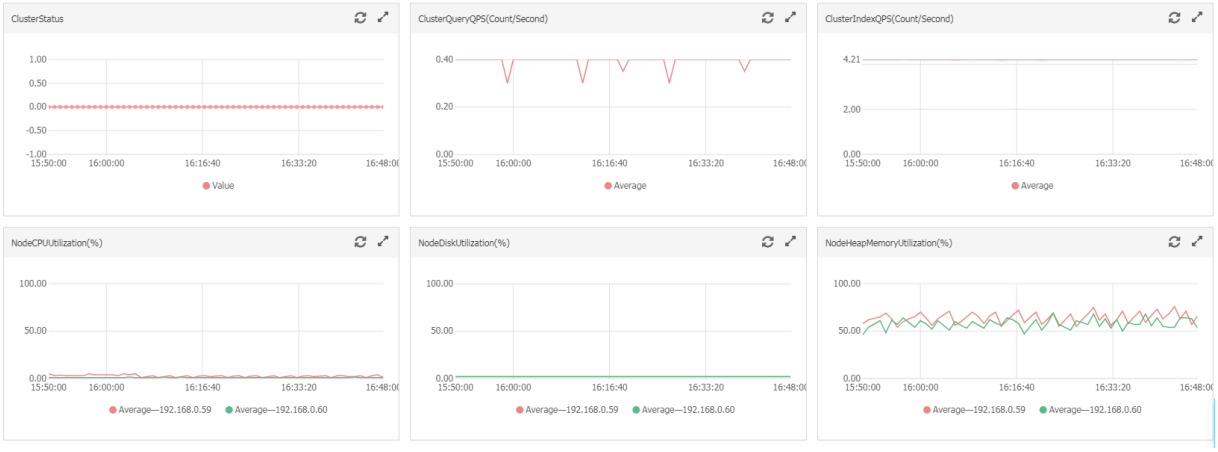
You can click Custom to specify the start time and end time to define a time window and view cluster monitoring data collected within the time window.



Note:

You can query up to 7 continuous days of data in the last 31 days by the minute.

Cluster monitoring metrics



1.7 Query logs

Alibaba Cloud Elasticsearch allows you to search and view multiple types of logs, including the Elasticsearch instance log, search slow log, indexing slow log, and GC log.

You can search for specific log entries by entering keywords and setting a time range . All Alibaba Cloud Elasticsearch log entries are sorted in time descending order. You can search for log entries that are stored within the last seven days.

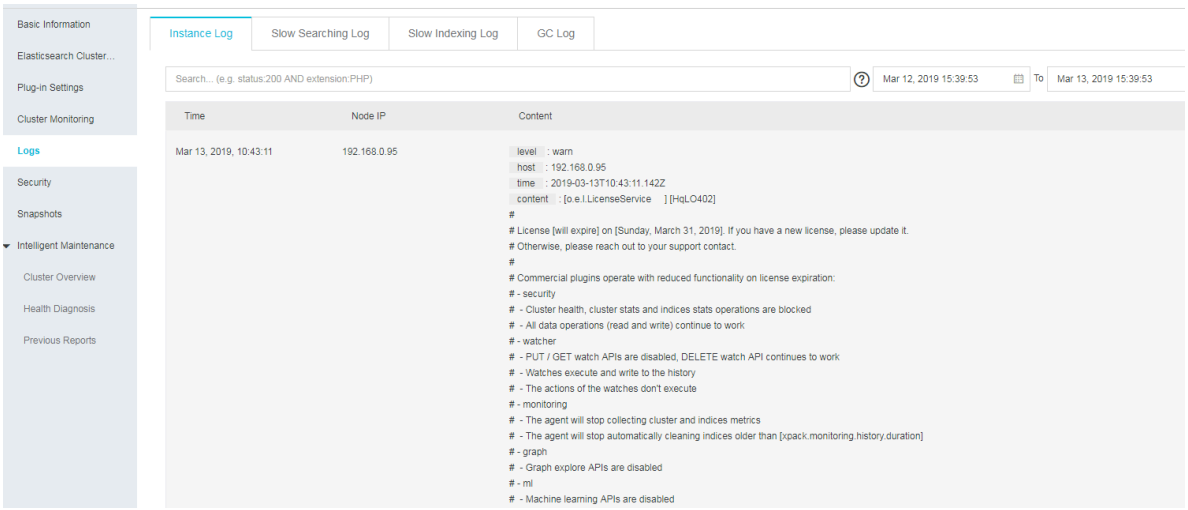
Alibaba Cloud Elasticsearch allows you to use Lucene to query logs. For more information, see *Query String Query*.

> **Note:**
> Due to the restrictions Elasticsearch puts on query conditions, a maximum of 10,000 log entries can be returned. If the log entries that you have queried are not contained in the returned 10,000 log entries, set a more specific time range to narrow down the search results.

Example

The following example shows how to search for Elasticsearch instance logs whose content contains the keyword `health`, level is set to `info`, and host is set to `192.168.1.123`.

1. Log on to the Alibaba Cloud Elasticsearch console, select the target instance, and click Manage in the Actions column to go to the Basic Information page. On the Basic Information page, click Logs in the left-side navigation pane and then click the Instance Log tab.
2. Enter `host:192.168.1.123 AND content:health AND level:info` in the search box.
3. Specify a time range and click Search.



Note:

- If you do not specify the end time, it defaults to the current system time.
- If you do not specify the start time, it defaults to one hour later than the end time.
- The word `AND` connecting search conditions that you enter in the search box must be capitalized.

Log description

You can view log entries that are retrieved based on specified search conditions on the log search page. Each log entry contains the following parts: Time, Node IP, and Content.

Time

The time when the log entry was created.

Node IP

The IP address of the Alibaba Cloud Elasticsearch node.

Content

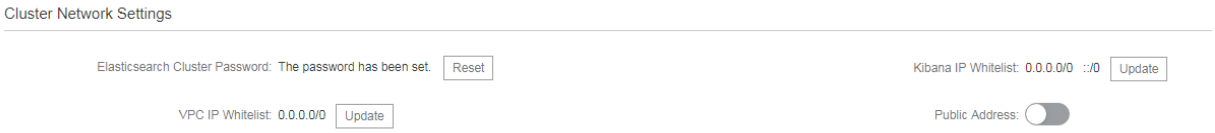
The information about the level, host, time, and content.

- level: the level of the log entry. Log levels include trace, debug, info, warn, and error. GC log entries do not have levels.
- host: indicates the IP address of the Elasticsearch node. You can view the IP address on the Nodes tab in the Kibana console.
- time: indicates the time when the log entry was created.
- content: displays major information about the log entry.

1.8 Security settings

Cluster network settings

You can reset the Elasticsearch cluster password, modify the Kibana IP whitelist and VPC IP whitelist, and enable public addresses and then configure the public IP whitelist.

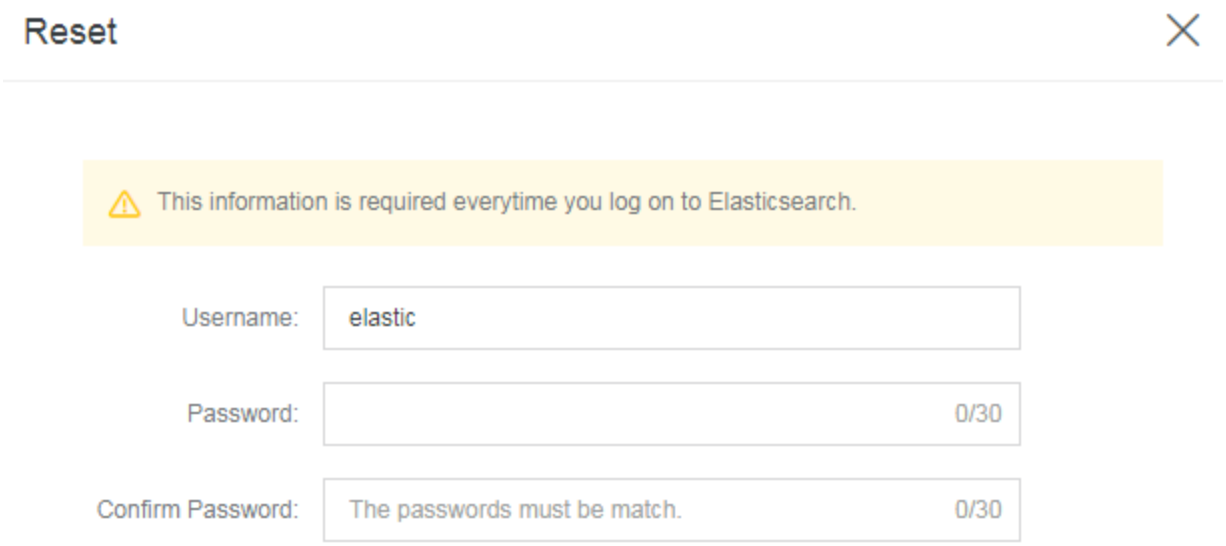


Elasticsearch cluster password

The password reset function resets the password of the administrator account elastic . After you reset the password, you can only use your new password to log on to the Kibana console and access Elasticsearch instances.

Note:

- The password reset operation does not change the password of non-elastic administrator accounts. We recommend that you do not use the elastic administrator account to access Elasticsearch instances.
- The new password will take effect 5 minutes after you submit the change.
- The password reset operation does not restart the corresponding Alibaba Cloud Elasticsearch instance.



Kibana IP whitelist

You can add comma-separated IP addresses or CIDR blocks to the Kibana IP whitelist, for example, `192 . 168 . 0 . 1` or `192 . 168 . 0 . 0 / 24` . Set the Kibana IP whitelist to `127 . 0 . 0 . 1` to forbid all IPv4 addresses. Set the Kibana IP whitelist to `0 . 0 . 0 . 0 / 0` to allow all IPv4 addresses.

Currently, only the China (Hangzhou) region supports using public IPv6 addresses to access Elasticsearch instances. This region also supports the Kibana IPv6 whitelist. You can add IPv6 addresses or CIDR blocks to the Kibana IPv6 whitelist, such as `2401 : b180 : 1000 : 24 :: 5` or `2401 : b180 : 1000 ::/ 48`. Set the Kibana IPv6 whitelist to `:: 1` to forbid all IPv6 addresses. Set the Kibana IPv6 whitelist to `::/ 0` to allow all IPv6 addresses.

Note:

By default, all public IP addresses are allowed to access Elasticsearch.

VPC IP whitelist

You can add comma-separated IP addresses or CIDR blocks to the VPC IP whitelist, for example, `192 . 168 . 0 . 1` or `192 . 168 . 0 . 1 / 24`. Set the VPC IP whitelist to `127 . 0 . 0 . 1` to forbid all IPv4 addresses. Set the VPC IP whitelist to `0 . 0 . 0 . 0 / 0` to allow all IPv4 addresses.

Note:

- By default, all VPC IPv4 addresses are allowed to access Elasticsearch.
- This whitelist is used to control access from VPCs to Elasticsearch.

Public addresses

Toggle the Public Address switch to green to enable the public address function. By default, this function is disabled.

Public IP address whitelist

You can add comma-separated IP addresses or CIDR blocks to the public IP address whitelist, for example, `192 . 168 . 0 . 1` or `192 . 168 . 0 . 1 / 24`. Set the public IP address whitelist to `127 . 0 . 0 . 1` to forbid all IPv4 addresses. Set the public IP address whitelist to `0 . 0 . 0 . 0 / 0` to allow all IPv4 addresses.

Currently, only the China (Hangzhou) region supports using public IPv6 addresses to access Elasticsearch instances. This region also supports the public IPv6 whitelist. You can add IPv6 addresses or CIDR blocks to the public IPv6 whitelist, such as `2401 : b180 : 1000 : 24 :: 5` or `2401 : b180 : 1000 ::/ 48`. Set the public IPv6 whitelist to `:: 1` to forbid all IPv6 addresses. Set the public IPv6 whitelist to `::/ 0` to allow all IPv6 addresses.

 **Note:**

By default, the public address function forbids all public IP addresses.

1.9 Configure synonyms for Elasticsearch instances

Configuration description

You can use a filter to configure synonyms. Sample code:

```
 PUT  / test_index
{
    " settings ": {
        " index " : {
            " analysis " : {
                " analyzer " : {
                    " synonym " : {
                        " tokenizer " : " whitespace ",

                        " filter " : [" synonym "]
                        }
                    },
                    " filter " : {
                        " synonym " : {
                            " type " : " synonym ",

                             " synonyms_p  ath " : " analysis /
 synonym . txt ",
                             " tokenizer " : " whitespace "

                        }
                    }
                }
            }
        }
}
```

- `filter` : Configure a `synonym` token filter that contains the path `analysis / synonym . txt` (relative to the config location).
- `tokenizer` : The tokenizer that controls the synonym, and is assigned to `whitespace` by default. Additional settings:
  - `ignore_cas  e` : Defaults to false.
  - `expand` : Defaults to true.

Currently, the synonym splitter supports the synonym formats Solr and WordNet:

- Solr synonyms

  Sample file formats:

  ```
  #  Blank  lines  and  lines  starting  with  pound  are
   comments .
  ```

```
# Explicit mappings match any token sequence on the
 LHS of "=>"
# and replace with all alternativ es on the RHS .
 These types of mappings
# ignore the expand parameter in the schema .
# Examples :
 i - pod , i pod => ipod ,
 sea biscuit , sea biscit => seabiscuit
# Equivalent synonyms may be separated with commas
 and give
# no explicit mapping . In this case the mapping
 behavior will
# be taken from the expand parameter in the schema
 . This allows
# the same synonym file to be used in different
 synonym handling strategies .
# Examples :
 ipod , i - pod , i pod
 foozball , foosball
 universe , cosmos
 lol , laughing out loud
# If expand == true , " ipod , i - pod , i pod " is
 equivalent
# to the explicit mapping :
 ipod , i - pod , i pod => ipod , i - pod , i pod
# If expand == false , " ipod , i - pod , i pod " is
 equivalent
# to the explicit mapping :
 ipod , i - pod , i pod => ipod
# Multiple synonym mapping entries are merged .
 foo => foo bar
 foo => baz
# is equivalent to
 foo => foo bar , baz
```

You can also directly define synonyms for the token filter in the configuration file. You must use `synonyms` instead of `synonyms_p ath` . Example:

```
 PUT / test_index
{
    " settings ": {
        " index " : {
            " analysis " : {
                " filter " : {
                    " synonym " : {
                        " type " : " synonym ",
                        " synonyms " : [
                            " i - pod , i pod => ipod ",
                            " begin , start "
                        ]
                    }
                }
            }
        }
    }
```

```
}
```

We recommend that you use `synonyms_p ath` to define large synonym sets in the file. Using `synonyms` to define large synonym sets will increase the size of your cluster.

- WordNet synonyms

  Synonyms based on the WordNet format can be declared using the following format:

```
 PUT  / test_index
{
    " settings ": {
        " index " : {
            " analysis " : {
                " filter " : {
                    " synonym " : {
                        " type " : " synonym ",
                        " format " : " wordnet ",
                        " synonyms " : [
                            " s ( 100000001 , 1 ,' abstain ', v , 1
 , 0 ).",
                            " s ( 100000001 , 2 ,' refrain ', v , 1
 , 0 ).",
                            " s ( 100000001 , 3 ,' desist ', v , 1 ,
 0 )."
                        ]
                    }
                }
            }
        }
    }
}
```

You can also use `synonyms_p ath` to define WordNet synonyms in a file.

Scenario A

Upload a synonym dictionary

1. Log on to the *Alibaba Cloud Elasticsearch console*.
2. Click create in the upper-left corner to create an Elasticsearch instance.
3. Click the instance to go to the configuration page.

4. Click Elasticsearch Cluster Configuration in the left-side menu bar, and click Upload Synonym Dictionary.

5. Click Upload, select the synonym dictionary (the txt files that are generated based on the rules you have set in the preceding steps) that you want to upload, and click Save.

   Wait until the Alibaba Cloud Elasticsearch instance takes effect and is in the normal status to use the synonym dictionary. This example uses `aliyun_synonyms.txt` as the test file, including: `begin, start`

Configure and test synonyms

1. Click Kinana console in the upper-right corner to go to the Kibana console of the Elasticsearch instance.
2. Click Dev tool on the left-side menu bar.
3. Run the following command in the console to create indexes:

```
PUT aliyun-index-test
{
"index": {
"analysis": {
"analyzer": {
"by_smart": {
"type": "custom",
"tokenizer": "ik_smart",
"filter": ["by_tfr","by_sfr"],
"char_filter": ["by_cfr"]
},
"by_max_word": {
"type": "custom",
"tokenizer": "ik_max_word",
"Filter": ["by_tfr", "by_sfr"],
"char_filter": ["by_cfr"]
}
},
"filter": {
"by_tfr": {
"type": "stop",
"stopwords": [" "]
},
"by_sfr": {
"type": "synonym",
```

```
        " synonyms_p  ath ": " analysis / aliyun_syn  onyms . txt "
      }
    },
    " char_filte  r ": {
      " by_cfr ": {
        " type ": " mapping ",
        " mappings ": ["| => |"]
      }
    }
  }
}
}
```

4. Run the following command to configure the synonym title fields:

```
 PUT   aliyun - index - test / _mapping / typename
{
" properties ": {
 " title ": {
   " type ": " text ",
   " index ": " analyzed ",
   " analyzer ": " by_max_wor  d ",
   " search_ana  lyzer ": " by_smart "
 }
}
}
```

5. Run the following commands to verify synonyms:

```
 GET   aliyun - index - test / _analyze
{
" analyzer ": " by_smart ",
" text ":" begin "
}
```

The following results will be returned if the configuration takes effect:

```
{
" tokens ": [
 {
   " token ": " begin ",
   " start_offs  et ":  0 ,
   " end_offset ":  2 ,
   " type ": " ENGLISH ",
   " position ":  0
 },
 {
   " token ": " start ",
   " start_offs  et ":  0 ,
   " end_offset ":  2 ,
   " type ": " SYNONYM ",
   " position ":  0
 }
]
}
```

6. Run the following command to add data and perform the following test:

```
 PUT   aliyun - index - test / doc / 1
{
" title ": " Shall   I   begin ?"
```

```
}
```

```
 PUT   aliyun - index - test / doc / 2
{
" title ": " I   start   work   at   nine ."
}
```

7. **Run the following command to test the query operation**

```
 GET   aliyun - index - test / _search
{
 " query " : { " match " : { " title " : " tomatoes " }},
 " highlight " : {
     " pre_tags " : ["< red >", "< bule >"],
     " post_tags " : ["</ red >", "</ bule >"],
     " fields " : {
         " title " : {
     }
 }
}
```

Normally, the following results will be returned:

```
{
" total ":  11 ,
" timed_out ":  false ,
" _shards " : {
 " total ":  5 ,
 " successful " :  5 ,
 " failed " :  0
},
" hits ": {
 " total ":  2
 " max_score ":  0 . 41048482 ,
 " hits ": [
   {
     " _index ": " aliyun - index - test ",
     " _type ": " doc ",
     " _id ": " 2 ",
     " _score ":  0 . 41048482 ,
     " source ": {
       " title ": " I   start   work   at   nine ."
     },
     " highlight ": {
       " title ": [
         " I  < red > start </ red >  work   at   nine ."
       ]
     }
   },
   {
     " _index ": " aliyun - index - test ",
     " _type ": " doc ",
     " _id ": " 1 ",
     " _score ":  0 . 39556286 ,
     " source ": {
       " title ": " Shall   I   begin ?"
     },
     " highlight ": {
       " title ": [
         " Shall   I  < red > begin </ red >?"
       ]
     }
```

```
    }
  ]
}
}
```

Scenario B

Follow these steps to directly reference the synonyms and use IK to filter the synonyms:

1. Configure a synonym filter `my_synonym _filter` and a synonym dictionary.
2. Configure an analyzer `my_synonym s` , and use `ik_smart` to separate words.

   After words are separated by `ik_smart` , lowercase all the letters and query synonyms.

```
 PUT  / my_index
{
 " settings ": {
    " analysis ": {
        " analyzer ": {
            " Maid  ":{
                " filter ": [
                    " lowercase ",
                    " my_synonym  _filter "
                ],
                " tokenizer ": " ik_smart "
            }
        },
        " filter ": {
            " my_synonym  _filter ": {
                " synonyms ": [
                    " begin , start "
                ],
                " type ": " synonym "
            }
        }
    }
 }
}
```

3. Run the following command to configure the synonym title fields:

```
 PUT  / my_index / _mapping / doc
{
" properties ": {
 " title ": {
   " type ": " text ",
   " index ": " analyzed ",
   " Analyzer ": " maid  ",
 }
}
}
```

4. Run the following commands to verify synonyms:

```
 GET  / my_index / _analyze
{
```

```
 " Analyzer ": " maid  ",
 " text ":" Shall   I   begin ?"
}
```

Normally, the following results will be returned:

```
{
" tokens ": [
 {
   " token ": " shall ",
   " start_offs  et ":  0 ,
   " end_offset ":  2 ,
   " type ": " ENGLISH ",
   " position ":  0
 },
 {
   " token ": " tomatoes ",
   " start_offs  et ":  0 ,
   " end_offset ":  2 ,
   " type ": " ENGLISH ",
   " position ":  0
 },
 {
   " token ": " begin ",
   " start_offs  et ":  0 ,
   " end_offset ":  2 ,
   " type ": " ENGLISH ",
   " position ":  0
 },
 {
   " token ": " start ",
   " start_offs  et ":  0 ,
   " end_offset ":  2 ,
   " type ": " SYNONYM ",
   " position ":  0
 }
]
}
```

5. **Run the following command to add data and perform the following test:**

```
 PUT  / my_index / doc / 1
{
" title ": " Shall   I   begin ?"
}
```

```
 PUT  / my_index / doc / 2
{
" title ": " I   start   work   at   nine ."
}
```

6. **Run the following command to test the query operation**

```
 GET  / my_index / _search
{
" query " : { " match " : { " title " : " tomatoes " }},
" highlight " : {
  " pre_tags " : ["< red >", "< bule >"],
  " post_tags " : ["</ red >", "</ bule >"],
  " fields " : {
      " title " : {
```

```
  }
}
}
```

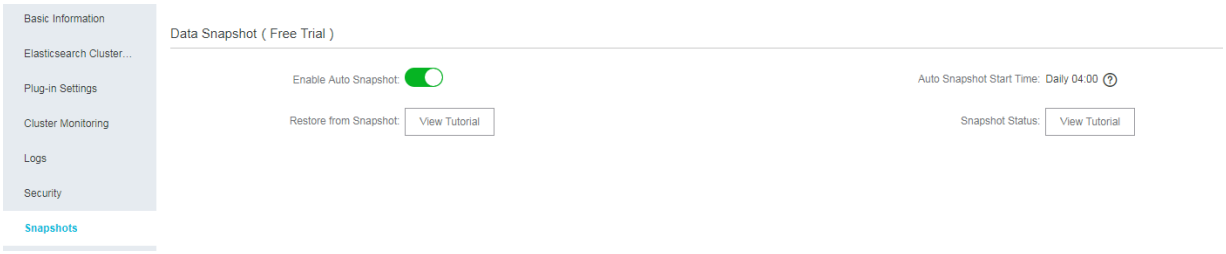
7. **Normally, the following results will be returned:**

```
{
" total ":  11 ,
" timed_out ":  false ,
" _shards " : {
 " total ":  5 ,
 " successful " :  5 ,
 " failed " :  0
},
" hits ": {
 " total ":  2
 " max_score ":  0 . 41913947 ,
 " hits ": [
   {
     " _index ": " my_index ",
     " _type ": " doc ",
     " _id ": " 2 ",
     " _score ":  0 . 41913947 ,
     " source ": {
       " title ": " I   start   work   at   nine ."
     },
     " highlight ": {
       " title ": [
         " I  < red > start </ red >  work   at   nine ."
       ]
     }
   },
   {
     " _index ": " my_index ",
     " _type ": " doc ",
     " _id ": " 1 ",
     " _score ":  0 . 39556286 ,
     " source ": {
       " title ": " Shall   I   begin ?"
     },
     " highlight ": {
       " title ": [
         " Shall   I  < red > begin </ red >?"
       ]
     }
   }
 ]
}
}
```

1.10 Data backup

1.10.1 Snapshots

Snapshots (free trial)



Enable auto snapshot

Toggle the Enable Auto Snapshot switch to green to enable the auto snapshot function. By default, this function is disabled.

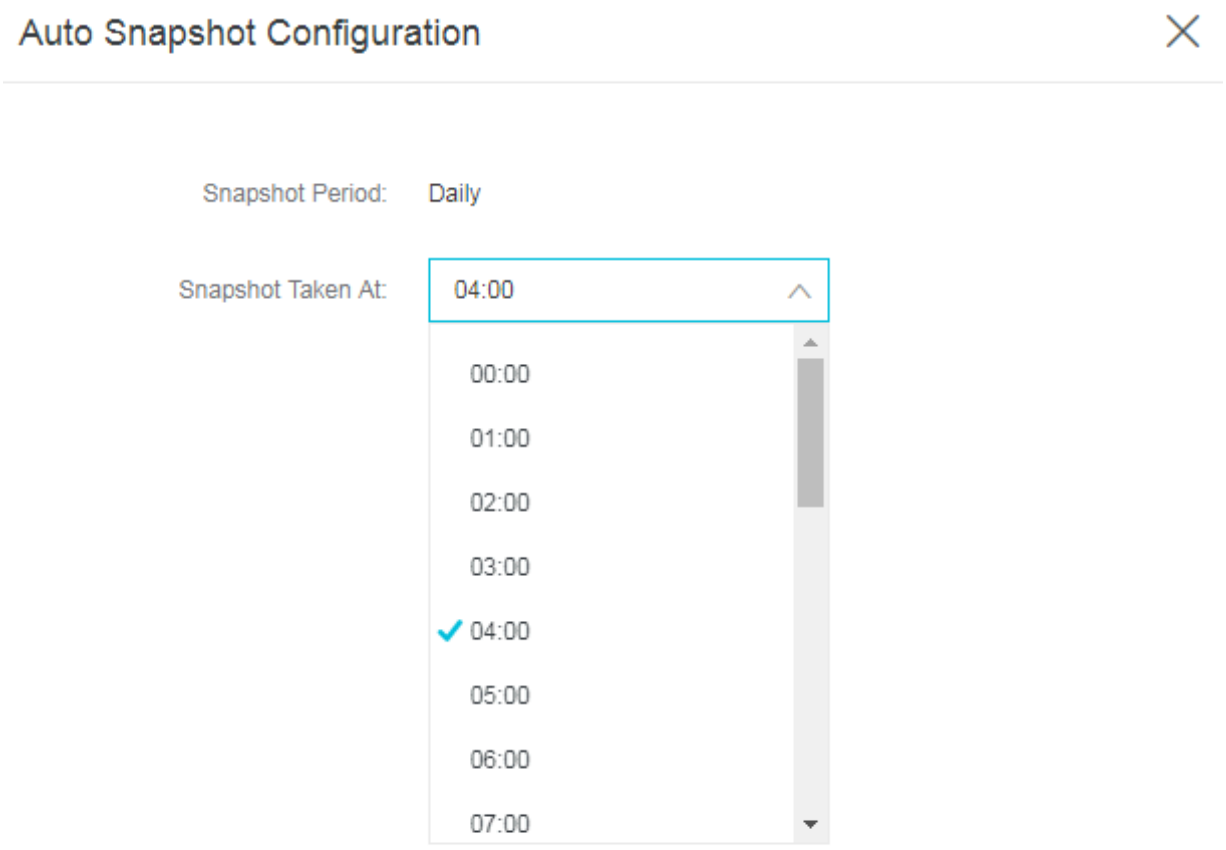
Auto snapshot start time

If the auto snapshot function is disabled, the system displays a message indicating that you must enable auto snapshot first.

> **Note:**
> If the auto snapshot function is enabled, the auto snapshot start time is set to the system time in the current region. Do not perform snapshot operations on the cluster when the system is creating snapshots.

Modify configuration

If the auto snapshot function is enabled, you can click Modify Configuration to change the auto snapshot start time.



Note:

- The auto snapshot period is set to daily.
- The auto snapshot start time is specified in hours. Valid values: [0-23].

Backup and recovery

For more information, click Backup and recovery.

Backup Status

For more information, click Backup and recovery.

1.10.2 View backup information

View automatic backup information

After enabling automatic backup, you can log on to the Kibana console that has been integrated into Alibaba Cloud Elasticsearch and run the Elasticsearch `snapshot` command in Dev Tools to view snapshots.

View all snapshots

Run the following command to view all the snapshots that are located in the aliyun_auto_snapshot repository.

```
GET _ snapshot / aliyun_aut o_snapshot / _ all
```

Response:

```
{
  " snapshots ": [
    {
      " snapshot ": " es - cn - abcdefghij klmn_20180 628092236 ",
      " uuid ": " n7YIayyZTm 2hwg8BeWby dA ",
      " version_id ": 5050399 ,
      " version ": " 2 . 0 . 0 ",
      " indices ": [
        ". kibana "
      ],
      " state ": " SUCCESS ",
      " start_time ": " 2018 - 06 - 28T01 : 22 : 39 . 609Z ",
      " start_time _in_millis ": 1530148959 609 ,
      " end_time ": " 2018 - 06 - 28T01 : 22 : 39 . 923Z ",
      " end_time_i n_millis ": 1530148959 923 ,
      " duration_i n_millis ": 314 ,
      " failures ": [],
      " _shards " : {
        " total ": 1
        " failed " : 0
        " successful ": 1 ,
      }
    },
    {
      " snapshot ": " es - cn - abcdefghij klmn_20180 628092500 ",
      " uuid ": " frdl1YFzQ5 Cn5xN9ZWuK LA ",
      " version_id ": 5050399 ,
      " version ": " 2 . 0 . 0 ",
      " indices ": [
        ". kibana "
      ],
      " state ": " SUCCESS ",
      " start_time ": " 2018 - 06 - 28T01 : 25 : 00 . 764Z ",
      " start_time _in_millis ": 1530149100 764 ,
      " end_time ": " 2018 - 06 - 28T01 : 25 : 01 . 482Z ",
      " end_time_i n_millis ": 1530149101 482 ,
      " duration_i n_millis ": 718 ,
      " failures ": [],
      " _shards " : {
        " total ": 1
        " failed " : 0
        " successful ": 1 ,
      }
    }
  ]
```

```
}
```

- state: Specifies the status of a snapshot. The snapshot status includes the following:
  - `IN_PROGRES S`: The snapshot is being restored.
  - `SUCCESS`: The snapshot has been restored and all shards have been successfully stored.
  - `FAILED`: The snapshot has been restored with an error. Some data cannot be stored.
  - `PARTIAL`: The snapshot has been successfully restored to an instance. However, one or more shards cannot be stored.
  - `INCOMPATIB LE`: The snapshot version is incompatible with the current instance version.

View specified snapshot

Run the following command to view detailed information about the specified snapshot in the aliyun_auto_snapshot repository.

```
GET _ snapshot / aliyun_aut o_snapshot /< snapshot >/ _ status
```

- `< Snapshot >`: Specifies the name of the snapshot, for example, `Es - cn - abcdefghij klmn_20180 628092236`.

Response:

```
{
  " Snapshots ": {
    {
      " snapshot ": " es - cn - abcdefghij klmn_20180 628092236 ",
      " repository ": " aliyun_aut o_snapshot ",
      " uuid ": " n7YIayyZTm 2hwg8BeWby dA ",
      " state ": " SUCCESS ",
      " shards_sta ts ": {
        " initializi ng ": 0 ,
        " started ": 0 ,
        " finalizing ": 0 ,
        " done ": 1 ,
        " failed " : 0
        " total ": 2
      },
      " stats ": {
        " number_of_ files ": 4 ,
        " processed_ files ": 4 ,
        " total_size _in_bytes ": 3296 ,
        " processed_ size_in_by tes ": 3296 ,
        " start_time _in_millis ": 1530148959 688 ,
        " time_in_mi llis ": 77
      },
      " indices ": {
        ". kibana ": {
```

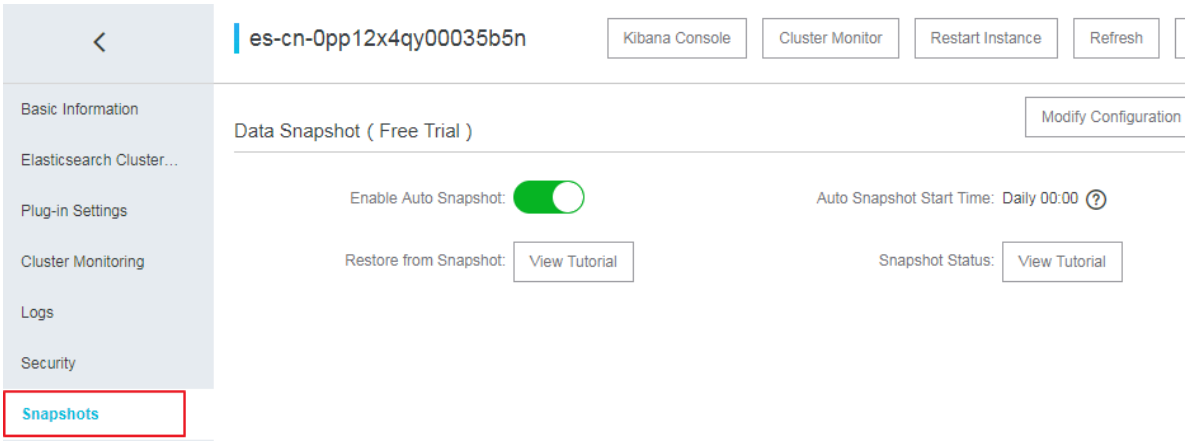
```
          " shards_sta  ts ": {
            " initializi  ng ":  0 ,
            " started ":  0 ,
            " finalizing ":  0 ,
            " done ":  1 ,
            " failed " :  0
            " total ":  2
          },
          " stats ": {
            " number_of_  files ":  4 ,
            " processed_  files ":  4 ,
            " total_size  _in_bytes ":  3296 ,
            " processed_  size_in_by  tes ":  3296 ,
            " start_time  _in_millis ":  1530148959  688 ,
            " time_in_mi  llis ":  77
          },
          " shards ": {
            " 0 ": {
              " stage ": " DONE ",
              " stats ": {
                " number_of_  files ":  4 ,
                " processed_  files ":  4 ,
                " total_size  _in_bytes ":  3296 ,
                " processed_  size_in_by  tes ":  3296 ,
                " start_time  _in_millis ":  1530148959  688 ,
                " time_in_mi  llis ":  77
              }
            }
          }
        }
      }
    }
  ]
}
```


1.10.3 Auto snapshot guide

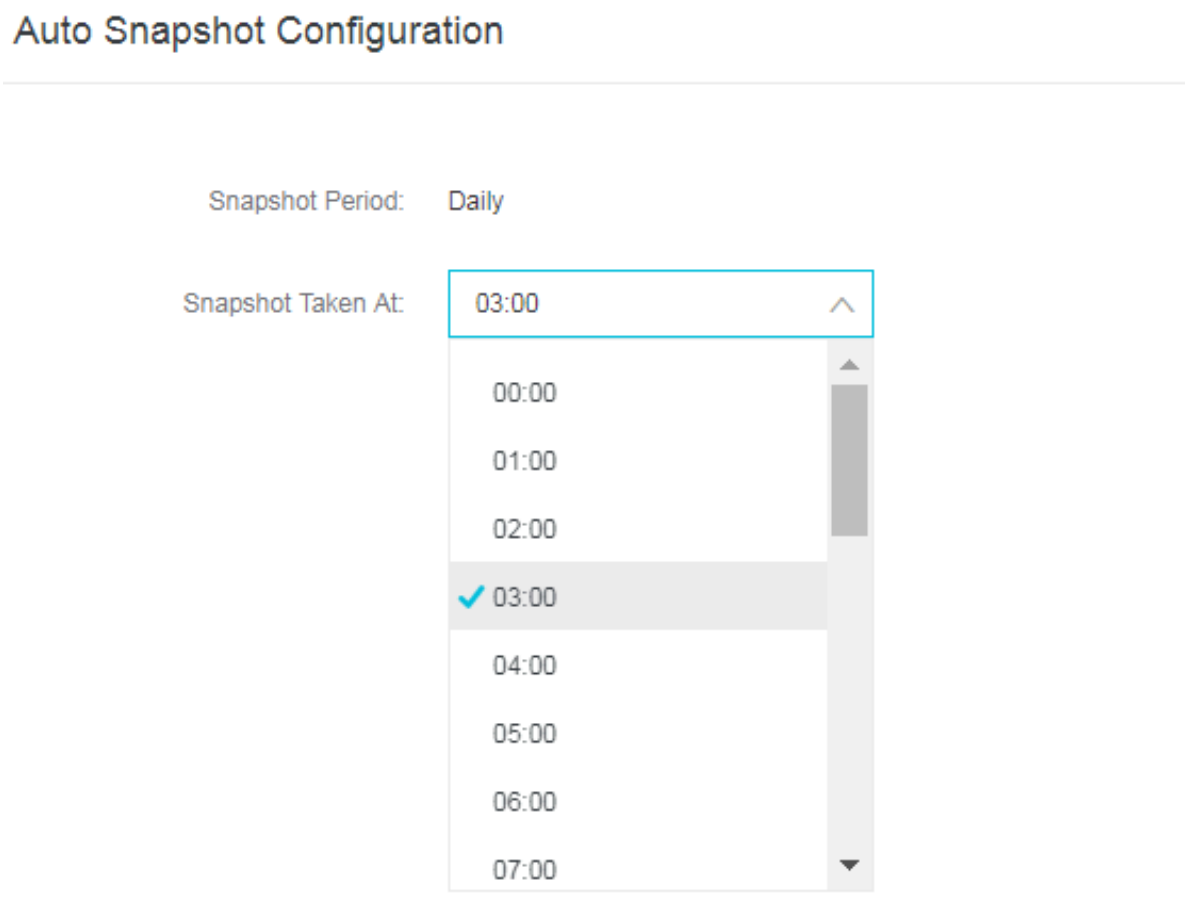
Enable auto snapshot

1. Log on to the Alibaba Cloud Elasticsearch console.
2. On the Instances page, click the target instance ID. You will be directed to the Basic Information page.
3. In the left-side navigation pane, click Snapshots.

4. On the Snapshots page, switch on Enable Auto Snapshot.



5. Click Modify Configuration in the upper-right corner to set the time when the daily snapshot is created.



Restore snapshots into instances

If you have enabled auto snapshot for a specified Alibaba Cloud Elasticsearch instance, snapshots will be automatically created on a daily basis. You can call the

corresponding `snapshot` operation to restore a snapshot into the Alibaba Cloud Elasticsearch instance where the snapshot is created.

 Note:

- The first snapshot is a complete backup created on a running Alibaba Cloud Elasticsearch instance. The following snapshots are created based on the incremental data of the Elasticsearch instance. Therefore, it takes a longer time to create the first snapshot, but a shorter time to create subsequent snapshots.
- Alibaba Cloud Elasticsearch only stores auto snapshots that are created within the last five days.
- A snapshot does not store monitoring data generated by an Alibaba Cloud Elasticsearch instance, such as the `. monitoring` and `. security_a udit` files.
- An auto snapshot can only be restored into the Alibaba Cloud Elasticsearch instance where the snapshot is created.
- An auto snapshot repository is created when the first snapshot is created.

View all snapshot repositories

You can run the following command to view all snapshot repositories:

```
GET   _snapshot
```

The following response is returned:

```
{
  " aliyun_aut  o_snapshot ": {
    " type ": " oss ",
    " settings ": {
      " compress ": " true ",
      " base_path ": " xxxx ",
      " endpoint ": " xxxx "
    }
  }
}
```

- aliyun_auto_snapshot: the name of the repository.
- type: the storage medium where snapshots are stored. This example uses Alibaba Cloud Object Storage Service (OSS).
- `compress : true` : enables compression of an index's metadata files.
- base_path: the location of the snapshots.
- endpoint: the region of the OSS instance.

Default parameters

Auto snapshots also support the following parameters that are not displayed:

- `max_snapsh ot_bytes_p er_sec : 40mb` : throttles per node snapshot rate. The default snapshot rate is 40 MB per second.
- `max_restor e_bytes_pe r_sec : 40mb` : throttles per node restore rate. The default restore rate is 40 MB per second.
- `chunk_size : Max 1Gb` : large files can be broken into smaller chunks during the snapshot process if needed. The maximum size of a chunk is 1 GB.

View all snapshots

You can run the following command to view all snapshots stored in the repository `aliyun_aut o_snapshot` :

```
GET _snapshot / aliyun_aut o_snapshot / _all
```

The following response is returned:

```
{
  " snapshots ": [
    {
      " snapshot ": " es - cn - abcdefghij klmn_20180 627091600 ",
      " uuid ": " MMRniVLPRA iawSCm8D8D ug ",
      " version_id ": 5050399 ,
      " version ": " 5 . 5 . 3 ",
      " indices ": [
        " index_1 ",
        ". security ",
        ". kibana "
      ],
      " state ": " SUCCESS ",
      " start_time ": " 2018 - 06 - 27T01 : 16 : 01 . 009Z ",
      " start_time _in_millis ": 1530062161 009 ,
      " end_time ": " 2018 - 06 - 27T01 : 16 : 05 . 632Z ",
      " end_time_i n_millis ": 1530062165 632 ,
      " duration_i n_millis ": 4623 ,
      " failures ": [],
      " shards ": {
        " total ": 12 ,
        " failed ": 0 ,
        " successful ": 12
      }
    }
  ]
}
```

Restore a snapshot into an instance

You can run the `_restore` command to restore a snapshot into an instance:

- Restore all indexes in a specified snapshot that is stored in the aliyun_aut o_snapshot repository. The restore tasks are executed in the background. `POST _snapshot / aliyun_aut o_snapshot /< snapshot >/ _restore`

  `< snapshot >`: replace it with the name of the specified snapshot. Example: `es - cn - abcdefghij klmn_20180 627091600`
- Restore all indexes in the specified snapshot that is stored in the aliyun_aut o_snapshot repository, and receive a response after all restore tasks are completed:

  The `_restore` command runs restore tasks asynchronously. The Alibaba Cloud Elasticsearch instance will return a response immediately if the restore command is executable. Restore tasks are executed in the background. You can add the `wait_for_c ompletion` parameter to the command. This parameter requires the Alibaba Cloud Elasticsearch instance to return the response only after the restore tasks are completed.

  `POST _snapshot / aliyun_aut o_snapshot /< snapshot >/ _restore ? wait_for_c ompletion = true`

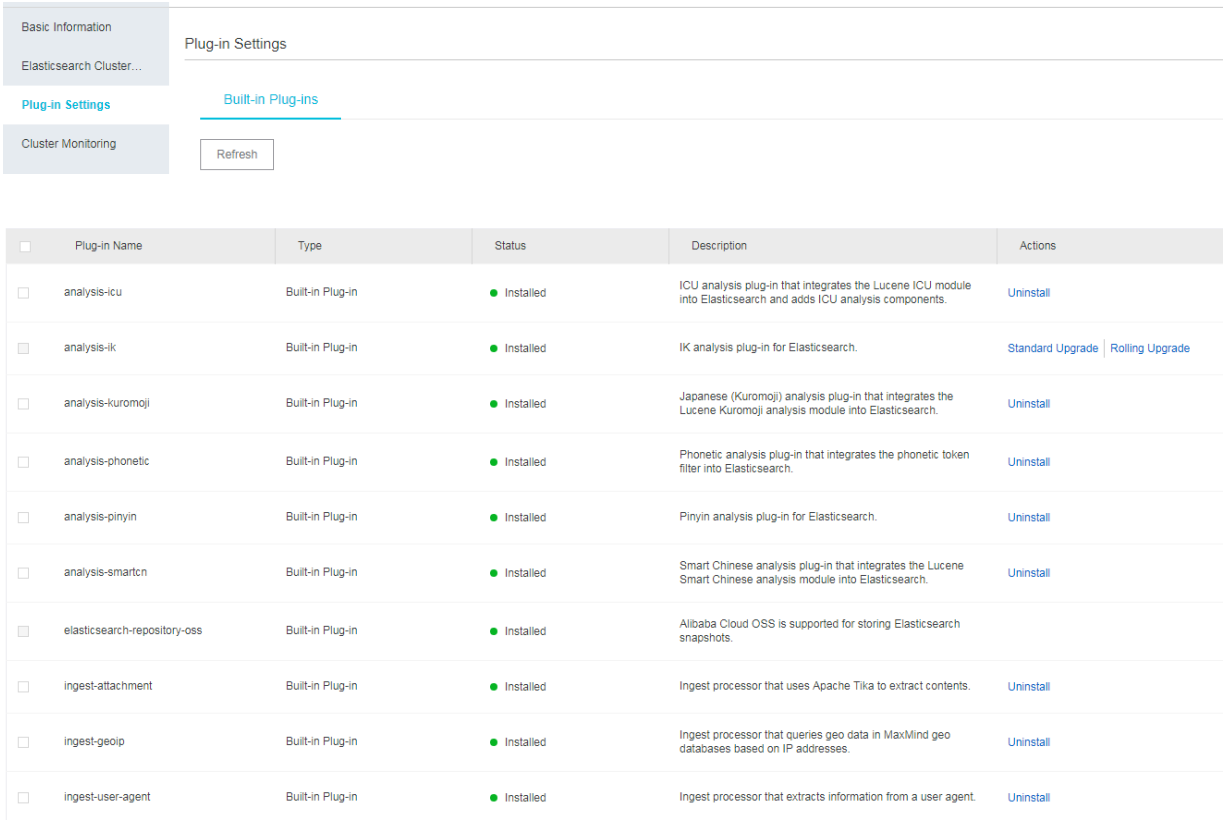
  `< snapshot >`: replace it with the name of the specified snapshot. Example: `es - cn - abcdefghij klmn_20180 627091600` .
- Restore indexes in the specified snapshot that is stored in the aliyun_aut o_snapshot repository, and rename the restored indexes. The restore tasks are executed in the background.

  ```
  POST _snapshot / aliyun_aut o_snapshot /< snapshot >/ _restore
  {
  " indices ": " index_1 ",
  " rename_pat tern ": " index_ (.+)",
  " rename_rep lacement ": " restored_i ndex_ $ 1 "
  }
  ```

  - `< snapshot >`: replace it with the name of the specified snapshot. Example: `es - cn - abcdefghij klmn_20180 627091600` .
  - indices: specifies names of the indexes that you need to restore.
  - rename_pattern: uses a regular expression to match the restored indexes. This parameter is optional.
  - rename_replacement: renames the index that matches the regular expression. This parameter is optional.

1.11 Plug-in settings

Built-in plug-ins



Upgrade IK analyzer dictionaries

Alibaba Cloud Elasticsearch allows you to use the following methods to update IK analyzer dictionaries:

- Standard upgrade
- Rolling upgrade

Standard upgrade

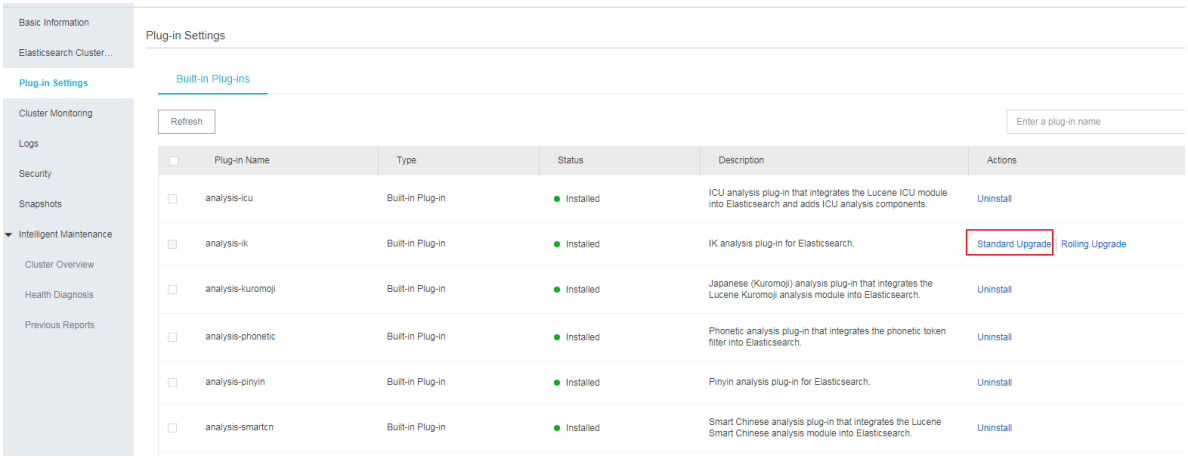
The standard upgrade method updates the dictionaries on all nodes in an Elasticsearch cluster. Elasticsearch will propagate the uploaded dictionary file to all Elasticsearch nodes in the cluster, modify the `IKAnalyzer . cfg . xml` file on the nodes, and restart the Elasticsearch nodes to load the dictionary file.

You can use the standard upgrade method to modify the IK main dictionary and stopword list. The standard upgrade page already shows the built-in main dictionary `SYSTEM_MAI N . dic` and stopword list `SYSTEM_STO PWORD . dic` .

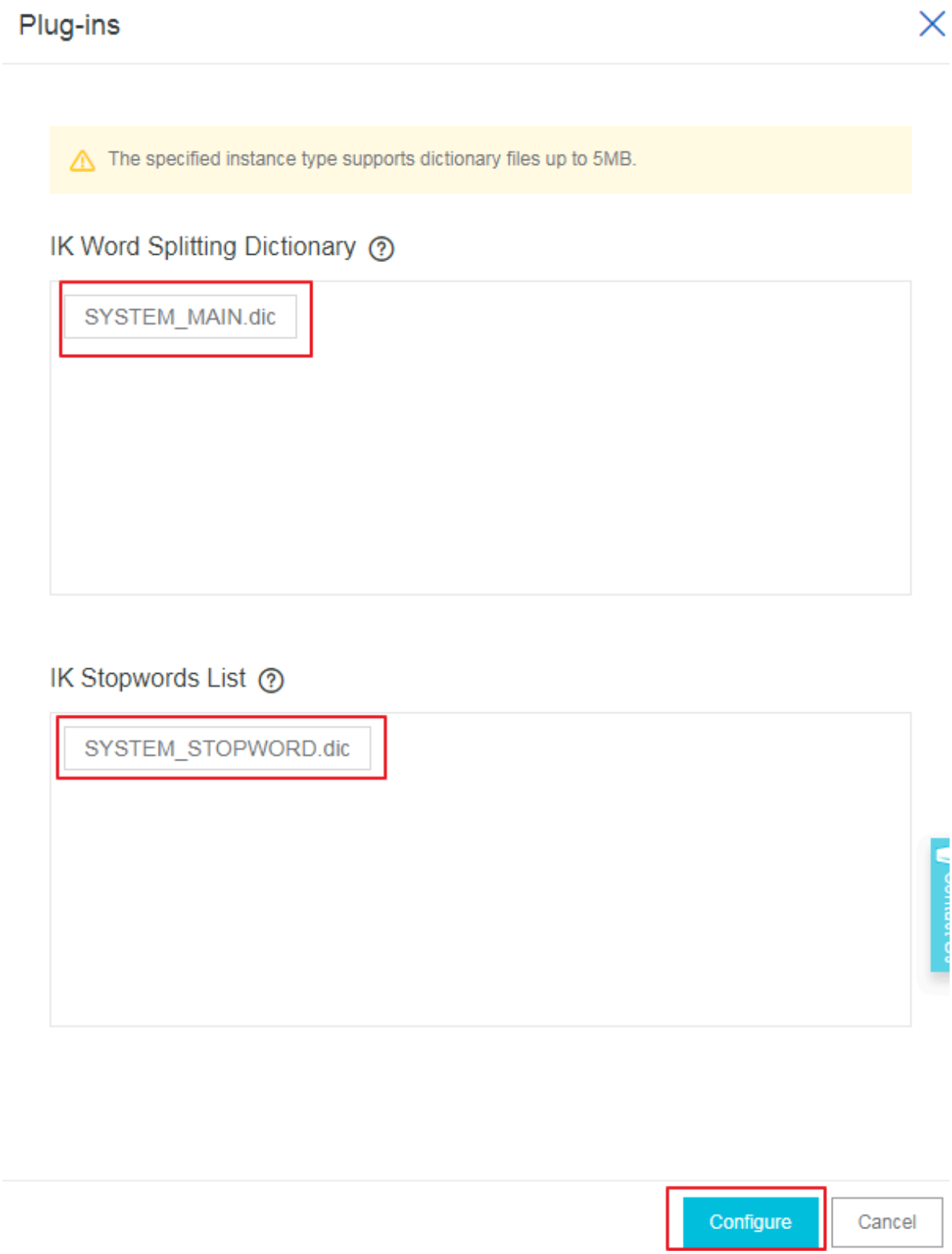
- To modify the built-in main dictionary, upload the `SYSTEM_MAIN.dic` dictionary.
- To modify the built-in stopword list, upload the `SYSTEM_STOPWORD.dic` dictionary.

Examples

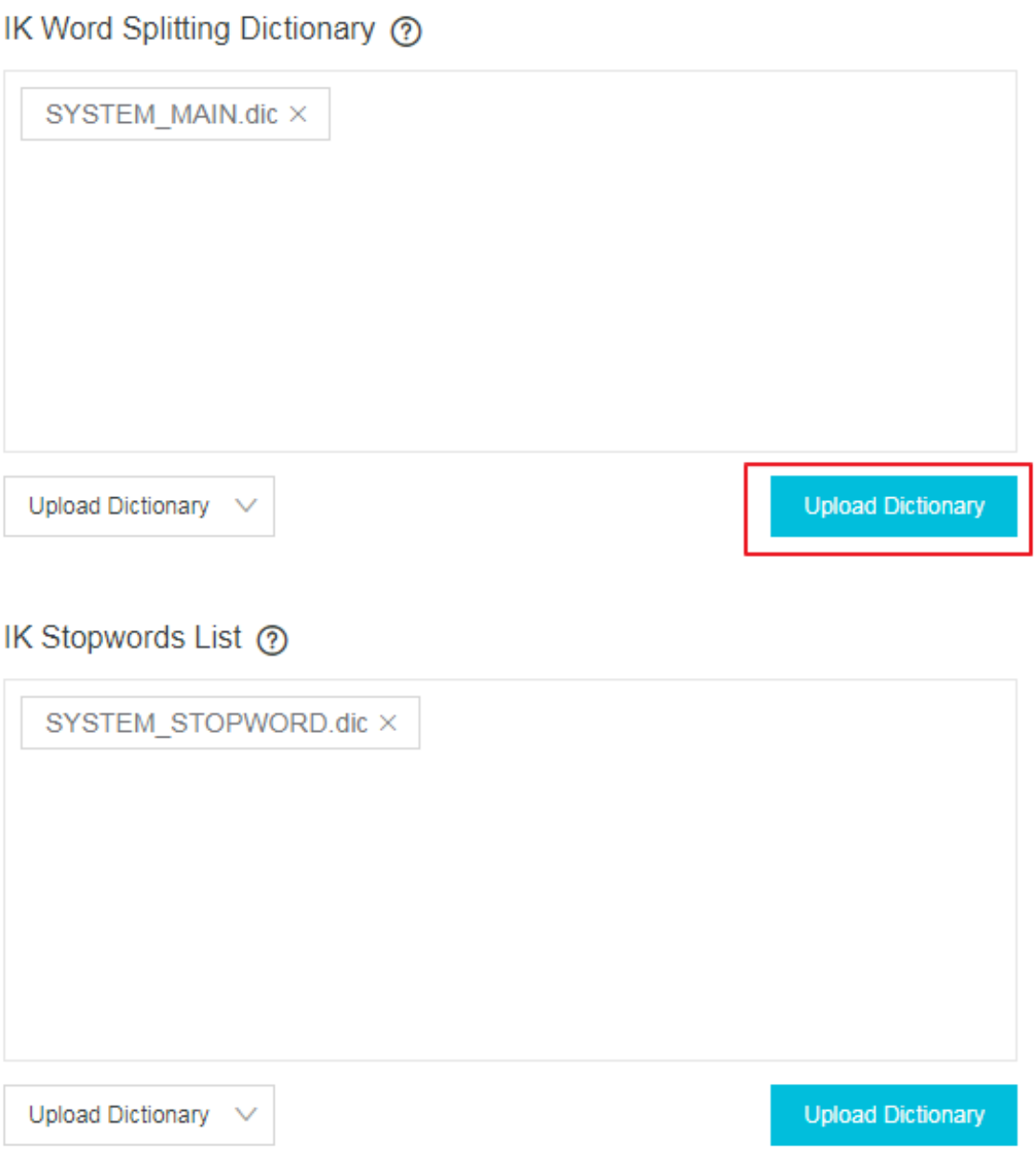
1. Log on to the Elasticsearch console, click the ID of the target Elasticsearch instance, click Plug-in Settings, and click Standard Upgrade for the IK analyzer.



2. Click Configure.



3. Click Upload Dictionary, and then upload a .dic main dictionary file.



Note:

- By default, you need to upload a dic file. You can also choose to import a dictionary file from OSS.
- To import a dictionary file from OSS, you need to modify the dictionary file on OSS and then import the file in the Elasticsearch console.

4. Go to the bottom of the page, select This operation requires a restart of the instance. Exercise with caution., and then click Save. The Elasticsearch cluster will restart all nodes.

5. After the cluster completes restarting all nodes in the cluster, log on to the Kibana console to verify the dictionary as follows:

```
 GET   _analyze
{
" analyzer ": " ik_smart ",
" text ": [" Words   in   your   dictionary "]
}
```

Note:

- You cannot delete the built-in IK main dictionary and stopword list.
- If you choose the standard upgrade method, both uploading a dictionary file and changing the dictionary content will restart all nodes in a cluster.
- You can perform the upgrade only when the Elasticsearch cluster is healthy.

Rolling upgrade

You can use the rolling upgrade method to update a dictionary when the content in the dictionary is changed. After you upload the new dictionary file, the Elasticsearch nodes will automatically load the new dictionary file.

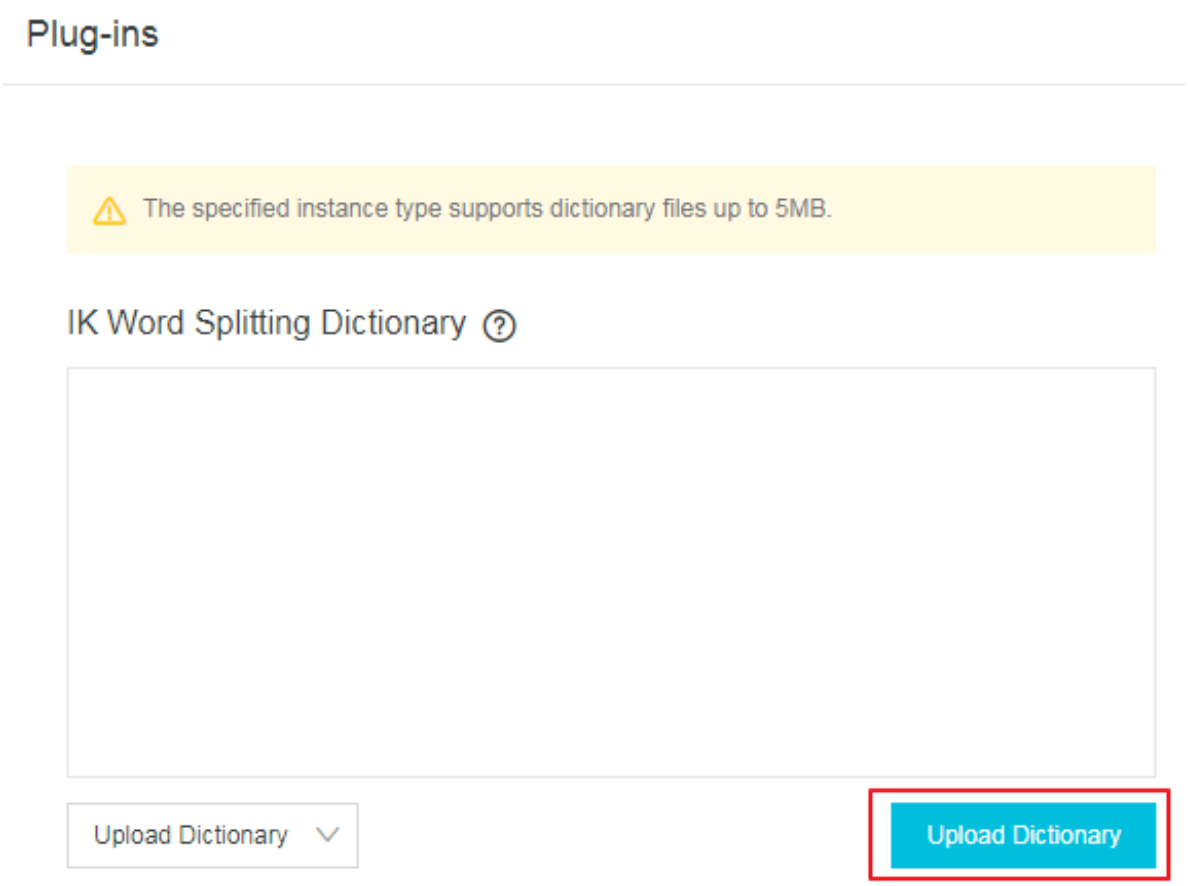
If you use the rolling upgrade method to update the dictionary list, such as uploading a new dictionary file or deleting a dictionary file, the `IKAnalyzer . cfg . xml` file will be modified. Therefore, you must restart all nodes in the cluster to reload the changes.

The procedure of updating a dictionary by using the rolling upgrade method is the same as that of the standard upgrade method. If this is the first time you upload a dictionary file, you must modify the `IKAnalyzer . cfg . xml` file and then restart all nodes in the cluster.

Examples

1. Log on to the Elasticsearch console, locate the target Elasticsearch instance, and click its ID. Click Plug-ins Settings, locate the IK plug-in, click Rolling Upgrade, and click Configure.

2. Click Upload Dictionary, and then select a main dictionary file.



Note:

- By default, you are required to upload a .dic file. You can also choose to import an OSS file.

- To import an OSS file, you must modify the file on OSS and then upload the file in the Elasticsearch console.

3. Go to the bottom of the page, select This operation requires a restart of the instance. Exercise with caution., and then click Save. The Elasticsearch cluster will restart all nodes. After the cluster has restarted all nodes, the uploaded dictionary takes effect.
4. To add more content to the dictionary or delete the content in the dictionary, modify the `a_10words . dic` file. Go to the rolling upgrade page, delete the `a_10words . dic` file, and upload the modified file. This operation only updates the content in an existing dictionary. Therefore, you do not need to restart all nodes in the cluster.
5. Go to the bottom of the page and then click Save. All nodes in the cluster will automatically load the modified dictionary file. It may take up to two minutes for all of the nodes to update the dictionary file. You must wait until all of the nodes finishing updating the dictionary file. To check whether the dictionary has been updated, call the following operation:

```
 GET   _analyze
{
" analyzer ": " ik_smart ",
" text ": [" Words   in   your   dictionary "]
}
```

 Note:

You cannot use the rolling upgrade method to modify the built-in dictionaries. To modify the built-in dictionaries, you must use the standard upgrade method.

2 ES self-built functions

Elasticsearch official documentation

Alibaba Cloud Elasticsearch is built based on open-source Elasticsearch 5.5.3. For more information, see *Elasticsearch Reference 5.5*.

SDK Client

The SDK client only supports HTTP requests. You can use *Java REST Client*, which is provided by Elasticsearch.

Elasticsearch Clients

- *Java REST Client [6.4] — other versions*
- *Java API [6.4] — other versions*
- *JavaScript API*
- *Groovy API [2.4] — other versions*
- *. NET API [6.x] — other versions*
- *PHP API [6.0] — other versions*
- *Perl API*
- *Python API*
- *Ruby API*
- *Community Contributed Clients*

3 Snapshot and recovery

You can use the `snapshot` API to back up your Alibaba Cloud Elasticsearch cluster. The API obtains the current status and data of your cluster and saves them to a shared repository. The backup process is "intelligent".

The first snapshot is a complete copy of data, and all subsequent snapshots only save the difference between existing snapshots and the new data. As you create snapshots for data from time to time, the backups are incrementally added and deleted. It means that the number of subsequent backups increases quite fast because only a very small data volume is transmitted.

 Notice:
Tags <1>, <2>, and <3> in the code in this article are used to mark the positions for convenient description on the code at the specified position. Remove these tags when running the code.

Create a repository

- OSS data sources of the standard storage type are recommended. OSS data sources of the archive storage type are not supported.
- <1> The OSS data source must be in the same region of your Alibaba Cloud Elasticsearch cluster. Enter the intranet address of the region in the `endpoint` field. For details, see the intranet endpoint for ECS access column in *Access domain name and data center*.
- <2> An OSS bucket must exist.

```
 PUT   _snapshot / my_backup
{
    " type ": " oss ",
    " settings ": {
        " endpoint ": " http :// oss - cn - hangzhou - internal .
 aliyuncs . com ", < 1 >
        " access_key  _id ": " xxxx ",
        " secret_acc  ess_key ": " xxxxxx ",
        " bucket ": " xxxxxx ", < 2 >
        " compress ":  true
    }
```

```
}
```

Define the shard size

If the size of the data to be uploaded is very large, we can define the shard size during snapshot. If the shard size is exceeded, the data is divided into multiple shards and then uploaded to the OSS instance.

- <1> Note that the `POST` instead of `PUT` method is set. In this way, the exiting repository settings are updated.
- <2> The start position of *`base_path`* to set the repository is the root directory by default.

```
 POST   _snapshot / my_backup / < 1 >
{
    " type ": " oss ",
    " settings ": {
        " endpoint ": " http :// oss - cn - hangzhou - internal .
 aliyuncs . com ",
        " access_key  _id ": " xxxx ",
        " secret_acc  ess_key ": " xxxxxx ",
        " bucket ": " xxxxxx ",
        " chunk_size ": " 500mb ",
        " base_path ": " snapshot /" < 2 >
    }
}
```

List the repository information

```
 GET   _snapshot
```

You can use `GET   _snapshot / my_backup` to obtain information of the specified repository.

Migrate backup snapshots

To migrate a snapshot to another cluster, you just need to back up the snapshot to the OSS instance register a snapshot respiratory (in the same OSS instance) on the new cluster, set the *`base_path`* to the position where the backup file is saved, and then execute the backup restoration command.

All opened indexes of a snapshot

A respiratory can contain multiple snapshots and each snapshot is related to a series of indexes, for example, all indexes, shard of indexes, or a single index. When creating a snapshot, specify an index for the snapshot and create a unique name for the snapshot.

Snapshot command

1. The following is a most basic snapshot command:

```
PUT _snapshot / my_backup / snapshot_1
```

This command is used to back up all opened indexes to the snapshot named `snapshot_1` in the respiratory `my_backup` . The call request is immediately returned and the snapshot is executed at the background.

2. If you want to wait until the execution finishes, add the tag `wait_for_completion` in the script.

```
PUT _snapshot / my_backup / snapshot_1 ? wait_for_c ompletion =
true
```

Then, the call is blocked until the snapshot execution finishes. If the snapshot size is large, it takes a long time to return the call request.

Specified indexes of a snapshot

All opened indexes are backed up by default. If Kibana is used and you do not want to back up all . `kibana` indexes related to diagnosis for disk space consideration,

you can back up only the specified indexes when creating snapshots for your cluster:

```
PUT _snapshot / my_backup / snapshot_2
{
    " indices ": " index_1 , index_2 "
}
```

When this snapshot command is run, only `index1` and `index2` are backed up.

List the snapshot information

Sometimes you may forget details about snapshots in the respiratory, especially when the snapshots are named according to the creation time, for example, `backup_201 4_10_28` .

1. You can initialize a `GET` request on a respiratory and snapshot name to obtain the information of a single snapshot:

```
GET _snapshot / my_backup / snapshot_2
```

The response contains all the details about the snapshot:

```
{
" snapshots ": [
  {
```

```
      " snapshot ": " snapshot_1 ",
      " indices ": [
         ". marvel_201  4_28_10 ",
         " index1 ",
         " index2 "
      ],
      " state ": " SUCCESS ",
      " start_time ": " 2014 - 09 - 02T13 : 01 : 43 . 115Z ",
      " start_time  _in_millis ":  1409662903  115 ,
      " end_time ": " 2014 - 09 - 02T13 : 01 : 43 . 439Z ",
      " end_time_i  n_millis ":  1409662903  439 ,
      " duration_i  n_millis ":  324 ,
      " failures ": [],
      " shards ": {
         " total ":  10 ,
         " failed ":  0 ,
         " successful ":  10
      }
   }
]
}
```

2. You can use the placeholder `_all` to replace the specific snapshot name to obtain a complete list of all snapshots in the respiratory:

```
GET   _snapshot / my_backup / _all
```

Delete a snapshot

You can initialize an HTTP-based call request on a respiratory/snapshot name through the `DELETE` API to delete all snapshots that are no longer used:

```
DELETE   _snapshot / my_backup / snapshot_2
```

It is important to delete a snapshot through the APIs. Other methods, such as manual deletion, are not supported. Snapshots increase incrementally and many snapshots may depend on historical snapshots. The `DELETE` API knows which data is still used by recent snapshots and thus only deletes snapshots that are no longer used.

Note:

If you delete backups manually, there is a risk that the backups may be seriously damaged because the deleted backups may contain data that is still being used.

Monitor the snapshot task progress

The `wait_for_c ompletion` tag provides the basic monitoring mode. If you want to restore the snapshots of a medium-sized cluster, this mode may be insufficient. The following two APIs provide more details about the snapshot status.

1. You can run a `GET` command for a snapshot ID to obtain the information of a specific snapshot:

```
GET _snapshot / my_backup / snapshot_3
```

If this command is run when the snapshot task is undergoing, you can view the start time, running time, and other information about the task.

Note:

This API uses the thread pool same as that of the snapshot mechanism. If the request is in a very large shard of the snapshot, the status update interval is large because the API is competing for resources in the same thread pool.

2. A better way is to drag the data through the `_status` API:

```
GET _snapshot / my_backup / snapshot_3 / _status
```

The following are detailed statistics returned by the `_status` API:

```
{
" snapshots ": [
  {
     " snapshot ": " snapshot_3 ",
     " repository ": " my_backup ",
     " state ": " IN_PROGRES  S ", < 1 >
     " shards_sta  ts ": {
        " initializi  ng ":  0 ,
        " started ":  1 , < 2 >
        " finalizing ":  0 ,
        " done ":  4 ,
        " failed " :  0
        " total ":  5
     },
     " stats ": {
        " number_of_  files ":  5 ,
        " processed_  files ":  5 ,
        " total_size  _in_bytes ":  1792 ,
        " processed_  size_in_by  tes ":  1792 ,
        " start_time  _in_millis ":  1409663054  859 ,
        " time_in_mi  llis ":  64
     },
     " indices ": {
        " index_3 ": {
           " shards_sta  ts ": {
              " initializi  ng ":  0 ,
              " started ":  0 ,
              " finalizing ":  0 ,
              " done ":  5 ,
              " failed ":  0 ,
              " total ":  5
           },
           " stats ": {
              " number_of_  files ":  5 ,
              " processed_  files ":  5 ,
              " total_size  _in_bytes ":  1792 ,
```

```
            " processed_  size_in_by  tes ":  1792 ,
            " start_time  _in_millis ":  1409663054  859 ,
            " time_in_mi  llis ":  64
         },
         " shards ": {
            " 0 ": {
               " stage ": " DONE ",
               " stats ": {
                  " number_of_  files ":  1 ,
                  " processed_  files ":  1 ,
                  " total_size  _in_bytes ":  514 ,
                  " processed_  size_in_by  tes ":  514 ,
                  " start_time  _in_millis ":  1409663054  862 ,
                  " time_in_mi  llis ":  22
               }
            },
            ...
```

- <1> The status of a running snapshot is IN_PROGRESS.
- <2> This specific snapshot still has a shard which is being uploaded. The other four shards have been uploaded.

  The response contains the overall information of the snapshot and statistics on each drilled-down index and shard. The following is a detailed figure about the snapshot task progress. Different shards of the snapshot can be in different states.

  INITIALIZING: The shard is checking the cluster status to see whether a snapshot can be created for it. This process is generally very fast.

  STARTED: The data is being transmitted to the repository.

  FINALIZING: Data transmission finishes and the shard is sending the snapshot metadata.

  DONE: The snapshot task is finished.

  FAILED: An error occurs when the snapshot is being processed, and the shard /index/snapshot task cannot be finished. You can check the log for more information.

Cancel a snapshot task

To cancel a snapshot task, you can run the following command when the task is running:

```
DELETE   _snapshot / my_backup / snapshot_3
```

The snapshot process is interrupted. The half-done snapshots in the respiratory are deleted.

Restoration from a snapshot

1. If you have backed up the data, you just need to add `_restore` to the ID of the snapshot which you want to restore to the cluster:

```
POST   _snapshot / my_backup / snapshot_1 / _restore
```

   All indexes saved in the snapshot are restored by default. If *snapshot_1* contains five indexes, all the five indexes are restored to the cluster. Like the `snapshot` API, you can select a specific index to be restored.

2. You can use an additional option to rename the indexes. The option allows you to use a mode to match the index name and rename the index through the restoration process. If you want to restore historical data to verify the content or perform other operations without replacing existing data, this option is very useful. The following is an example about how to restore a single index from a snapshot and rename the index:

```
 POST  / _snapshot / my_backup / snapshot_1 / _restore
{
 " Indices ": " index_1 ", < 1 >
 " rename_pat  tern ": " index_ (.+)", < 2 >
 " rename_rep  lacement ": " restored_i  ndex_ $ 1 " < 3 >
}
```

   The `index_1` is restored to your cluster but is renamed to `restored_i  ndex_1` .

   - <1> Only `index_1` is restored. Other indexes in the snapshot are neglected.
   - <2> Search for indexes being restored that can match the provided mode.
   - <3> Rename the indexes to the alternative ones.

3. Like the snapshot, the `restore` command is immediately returned and the restoration process runs in the background. If you want your HTTP call request to be blocked until the restoration process is finished, add the `wait_for_c ompletion` tag:

```
POST   _snapshot / my_backup / snapshot_1 / _restore ?  wait_for_c
ompletion = true
```

Monitor the restoration operation

The existing restoration mechanism of Elasticsearch is referenced for restoring data from the respiratory. As for internal realization, shard restoration from a respiratory is equivalent to restoration from another node.

To monitor the restoration progress, call the `recovery` API. This API is for general purpose and is used to display the status of moving shards in your cluster.

1. **This API can be used to independently call a specified restored index:**

```
GET   restored_i  ndex_3 / _recovery
```

2. **It can also be used to view all indexes in your cluster, including moving indexes that are unrelated to the restoration process:**

```
GET  / _recovery /
```

The following is an output example. Note that a large quantity of content may be output if your cluster is highly active

```
{
" restored_i  ndex_3 " : {
 " shards " : [ {
   " id " :  0 ,
   " type " : " snapshot ", < 1 >
   " stage " : " index ",
   " primary " :  true ,
   " start_time " : " 2014 - 02 - 24T12 : 15 : 59 . 716 ",
   " stop_time " :  0 ,
   " total_time  _in_millis " :  175576 ,
   " source " : { < 2 >
     " repository " : " my_backup ",
     " snapshot " : " snapshot_3 ",
     " index " : " restored_i  ndex_3 "
   },
   " target " : {
     " id " : " ryqJ5lO5S4 - lSFbGntkEk  g ",
     " hostname " : " my . fqdn ",
     " ip " : " 10 . 0 . 1 . 7 ",
     " name " : " my_es_node "
   },
   " index " : {
     " files " : {
       " total " :  73 ,
       " reused " :  0 ,
       " recovered " :  69 ,
       " percent " : " 94 . 5 %" < 3 >
     },
     " bytes " : {
       " total " :  79063092 ,
       " reused " :  0 ,
       " recovered " :  68891939 ,
       " percent " : " 87 . 1 %"
     },
     " total_time  _in_millis " :  0
   },
   " translog " : {
     " recovered " :  0 ,
     " total_time  _in_millis " :  0
   },
   " start " : {
     " check_inde  x_time " :  0 ,
     " total_time  _in_millis " :  0
```

```
    }
  } ]
}
}
```

- <1> The `type` field indicates the restoration type. The shard is restored from a snapshot.
- <2> The `source` field indicates the snapshot and repository from which the shard is restored.
- <3> The `percent` field indicates the restoration progress. The specified shard is restored by 94% until now. The restoration task will soon be finished. The specified shard is restored by 94% until now. The restoration task will soon be finished.

All indexes under restoration and all shards in these indexes are listed in the output . Statistics on the start/end time, duration, restoration progress in percentage, and number of transmitted bytes, of each shard are displayed.

Cancel a restoration task

You can delete an index being restored to cancel a restoration task. You can modify the cluster status by calling the `DeletIndex` API to stop a restoration process. For example:

```
DELETE  / restored_i  ndex_3
```

If `restored_i  ndex_3` is being restored, after this deletion command is run, the restoration process stops and all the data that has been restored to the cluster is deleted.

4 RAM

4.1 Authorized resources

The following table lists the supported resource types and the corresponding Aliyun resource names (ARN).

Resource type	ARN
instances	acs:elasticsearch:$regionId:$accountId:instances/$instanceId
user	acs:elasticsearch:$regionId:$accountId:user/alert

- $regionId: the ID of the specified region. You can also enter an asterisk `(*)`.
- $accountId: the ID of your Alibaba Cloud account. You can also enter an asterisk `(*)`.
- $instanceId: the ID of a specified Alibaba Cloud Elasticsearch instance. You can also enter an asterisk `(*)`.
- You can replace the word alert in the user ARN with an asterisk `(*)` .

Instance authorization

Note:

The following ARNs are shortened. For the complete name information, see the preceding table.

- Common actions on instances

Action	Description	ARN
elasticsearch:CreateInstance	You can perform this action to create an instance.	`instances /*`
elasticsearch:ListInstance	You can perform this action to view instances.	`instances /*`
elasticsearch:DescribeInstance	You can perform this action to view instance description.	`instances /*` or `instances /$instanceId`

Action	Description	ARN
elasticsearch:DeleteInstance	You can perform this action to delete an instance.	`instances /*` or `instances /$instanceId`
elasticsearch:RestartInstance	You can perform this action to restart an instance.	`instances /*` or `instances /$instanceId`
elasticsearch:UpdateInstance	You can perform this action to update an instance.	`instances /*` or `instances /$instanceId`

- Actions on plug-ins

Action	Description	ARN
elasticsearch:ListPlugin	You can perform this action to obtain the list of plug-ins.	`instances /$instanceId`
elasticsearch:InstallSystemPlugin	You can perform this action to install system plug-ins.	`instances /$instanceId`
elasticsearch:UninstallPlugin	You can perform this action to install a plug-in.	`instances /$instanceId`

- Actions on networks

Action	Description	ARN
elasticsearch:UpdatePublicNetwork	You can perform this action to check whether access through the public address is allowed.	`instances /$instanceId`
elasticsearch:UpdatePublicIps	You can perform this action to modify the public network whitelist.	`instances /$instanceId`
elasticsearch:UpdateWhiteIps	You can perform this action to modify the VPC whitelist.	`instances /$instanceId`
elasticsearch:UpdateKibanaIps	You can perform this action to modify the Kibana whitelist.	`instances /$instanceId`

- Actions on dictionaries

Action	Description	ARN
elasticsearch:UpdateDict	You can perform this action to modify the IK analyzer and synonym dictionary.	`instances /$ instanceId`

User authorization

> Note:
> The following ARNs are shortened. For the full names, see the table in the first section.

- Authorized CloudMonitor actions (Elasticsearch console)

> Note:
> You must activate CloudMonitor before you perform the DescribeAlert action.

Action	Description	ARN
elasticsearch:DescribeAlert	You can perform this action to check if Quick Alarm is enabled.	`user /*` or `user / alert`

- Authorized CloudMonitor actions (CloudMonitor console)

Action	Description	ARN
cms:ListProductOfActiveAlert	You can perform this action to view services that have CloudMonitor enabled.	*
cms:ListAlarm	You can perform this action to query the specified or all alarm rule settings.	*
cms:QueryMetricList	You can perform this action to query the monitoring data of a specified instance.	*

Intelligent Maintenance authorization

 **Note:**

The following ARNs are shortened. For the full names, see the table in the first section.

Action	Description	ARN
elasticsearch:OpenDiagnosis	You can perform this action to enable health diagnosis.	`instances /*` or `instances /$instanceId`
elasticsearch:CloseDiagnosis	You can perform this action to disable health diagnosis.	`instances /*` or `instances /$instanceId`
elasticsearch:UpdateDiagnosisSettings	You can perform this action to update the health diagnosis settings.	`instances /*` or `instances /$instanceId`
elasticsearch:DescribeDiagnosisSettings	You can perform this action to query the health diagnosis settings.	`instances /*` or `instances /$instanceId`
elasticsearch:ListInstanceIndices	You can perform this action to query instance indexes.	`instances /*` or `instances /$instanceId`
elasticsearch:DiagnoseInstance	You can perform this action to start health diagnosis	`instances /*` or `instances /$instanceId`
elasticsearch:ListDiagnoseReportIds	You can perform this action to query diagnosis report IDs.	`instances /*` or `instances /$instanceId`
elasticsearch:DescribeDiagnoseReport	You can perform this action to view diagnosis report details.	`instances /*` or `instances /$instanceId`
elasticsearch:ListDiagnoseReport	You can perform this action to list diagnosis reports.	`instances /*` or `instances /$instanceId`

Supported regions

Elasticsearch region	RegionId
China (Hangzhou)	cn-hangzhou
China (Beijing)	cn-beijing
China (Shanghai)	cn-shanghai
China (Shenzhen)	cn-shenzhen
India (Mumbai)	ap-south-1
Singapore	ap-southeast-1
China (Hong Kong)	cn-hongkong
US (Silicon Valley)	us-west-1
Malaysia (Kuala Lumpur)	ap-southeast-3
Germany (Frankfurt)	eu-central-1
Japan (Tokyo)	ap-northeast-1
Australia (Sydney)	ap-southeast-2
Indonesia (Jakarta)	ap-southeast-5

4.2 Access authentication rules

General permission policies

The following two general permission policies are provided to meet the needs for common access, so that you can select a permission policy suitable for you. You can search for the policy name in the brackets from Optional Authorization Policy Names and select it.

- Read-only permissions for Elasticsearch instances, applicable for read-only users (AliyunElasticsearchReadOnlyAccess).
- Administrator permissions for Elasticsearch instances, applicable for the administrator (AliyunElasticsearchFullAccess).

 Note:
If none of the above general permission policies can meet your needs, you can refer to the following description and customize a permission policy.

Permission to buy instances (post-payment & prepayment)

Permission to access the VPC of the primary account

- [“vpc:DescribeVSwitch*” , “vpc:DescribeVpc*”]

> Note:
> You can refer to the system template AliyunVPCReadOnlyAccess.

Subaccount order permission

- [“bss:PayOrder”]

> Note:
> You can refer to the system template AliyunBSSOrderAccess.

API permissions

Method	URI	Resource	Action
GET	/instances	instances/*	ListInstance
POST	/instances	instances/*	CreateInstance
GET	/instances/$instanceId	instances/$instanceId	DescribeInstance
DELETE	/instances/$instanceId	instances/$instanceId	DeleteInstance
POST	/instances/$instanceId/actions/restart	instances/$instanceId	RestartInstance
PUT	/instances/$instanceId	instances/$instanceId	UpdateInstance

Authorization examples

- *Authorized resources* (for example, $regionid, $accountid, and $instanceId).
- Elasticsearch instances in the resource can be indicated by the wildcard `*`.

Authorization example 1

To a subaccount under the primary account (accountId “1234”), assign all operation permissions, except for CreateInstance, over all instances in China East 1 (Hangzhou) on the console, and set the instances to be accessible from only the specified IP address.

After this policy is created on the console of the primary account, you need to use your primary account on the RAM console or the RAM SDK to authorize the subaccount.

1. Create a policy

```
{
  " Statement  ":[
    {
      " Action ": [
        " imagesearc  h : ListInstan  ce ",
        " imagesearc  h : DescribeIn  stance ",
        " elasticsea  rch : DeleteInst  ance ",
        " elasticsea  rch : RestartIns  tance ",
        " elasticsea  rch : UpdateInst  ance "
      ],
      " Condition ": {
        " IpAddress ": {
          " acs : SourceIp ": " xxx . xx . xxx . x / xx "
        }
      },
      " Effect ": " Allow ",
      " Resource ": " acs : imagesearc  h : cn - shanghai : 1234 :
 instance /*"
    }
  ],
  " Version ": " 1 "
}
```

2. Authorize the current policy to your specified subaccount.

Authorization example 2

For a subaccount under the primary account (accountId “1234”), assign all operation permissions, except for CreateInstance, over the specified instances in China East 1 (Hangzhou) on the console, and set the instances to be accessible from only the specified IP address.

After this policy is created on the console of the primary account, you should authorize the subaccount through your primary account on the RAM console or use RAM SDK to authorize the subaccount.

1. Create a policy

```
{
  " Statement  ":[
    {
      " Action ": [
        " elasticsea  rch : ListInstan  ce "
      ],
      " Condition ": {
        " IpAddress ": {
          " acs : SourceIp ": " xxx . xx . xxx . x / xx "
        }
      },
```

```
      " Effect ": " Allow ",
      " Resource ": " acs : imagesearc  h : cn - shanghai : 1234 :
 instance /*"
    },
    {
      " Action ": [
        " elasticsea  rch : DescribeIn  stance ",
        " elasticsea  rch : DeleteInst  ance ",
        " elasticsea  rch : RestartIns  tance ",
        " elasticsea  rch : UpdateInst  ance "
      ],
      " Condition ": {
        " IpAddress ": {
          " acs : SourceIp ": " xxx . xx . xxx . x / xx "
        }
      },
      " Effect ": " Allow ",
      " Resource ": " acs : elasticsea  rch : cn - hangzhou : 1234 :
 instances /$ instanceId "
    }
  ],
  " Version ": " 1 "
}
```

2. Authorize the current policy to your specified subaccount.

Authorization example 3

To a subaccount under the primary account (accountId “1234”), assign all operation permissions over all instances in all regions supported by Alibaba Cloud Elasticsea rch on the console.

After this policy is created on the console of the primary account, you should authorize the subaccount through your primary account on the RAM console or use RAM SDK to authorize the subaccount.

1. Create a policy

```
{
  " Statement  ":[
    {
      " Action ": [
          " elasticsea  rch :*"
          ],
      " Effect ": " Allow ",
      " Resource ": " acs : imagesearc  h :*: 1234 : instance /*"
    }
  ],
  " Version ": " 1 "
}
```

2. Authorize the current policy to your specified subaccount.

Authorization example 4

To a subaccount under the primary account (accountId “1234”), assign all operation permissions, except for CreateInstance and ListInstance, over specified instances in all regions supported by Alibaba Cloud Elasticsearch on the console.

After this policy is created on the console of the primary account, you should authorize the subaccount through your primary account on the RAM console or use RAM SDK to authorize the subaccount.

1. Create a policy

```
{
  " Statement  ":[
    {
      " Action ": [
          " elasticsea  rch : DescribeIn  stance ",
          " elasticsea  rch : DeleteInst  ance ",
          " elasticsea  rch : UpdateInst  ance ",
          " elasticsea  rch : RestartIns  tance "
            ],
      " Effect ": " Allow ",
      " Resource ": " acs : elasticsea  rch :*: 1234 : instances /$
 instanceId "
    }
  ],
  " Version ": " 1 "
}
```

2. Authorize the current policy to your specified subaccount.

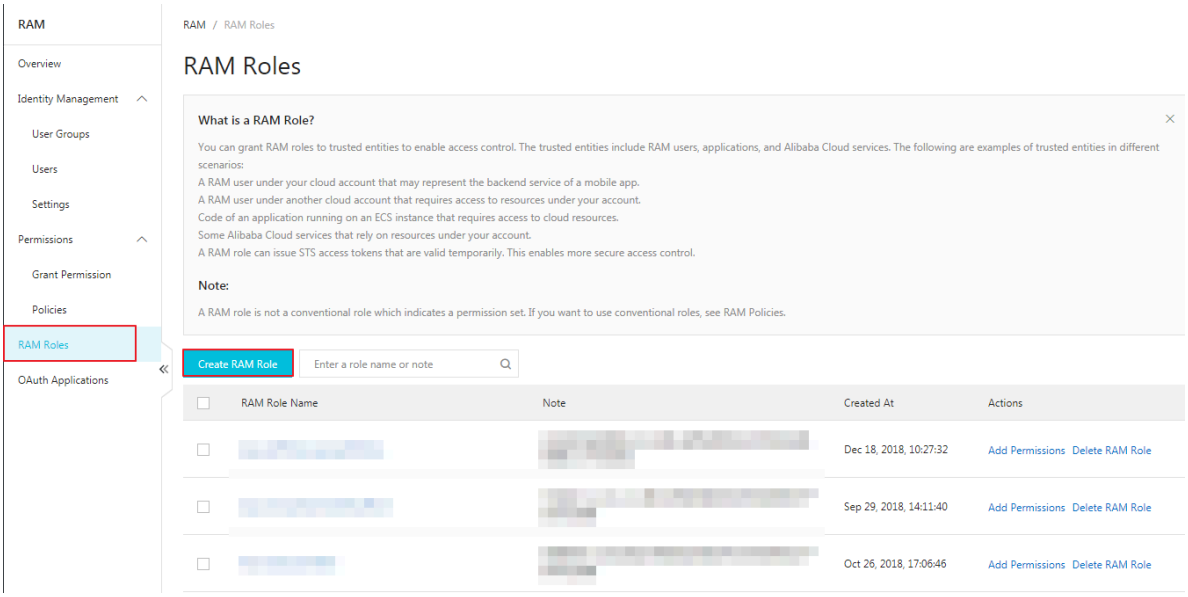
4.3 Temporary access token

Users (people or applications) that only access your cloud resources occasionally are called temporary users. You can use Security Token Service (STS, an extended authorization service of RAM) to issue an access token to these users (subaccounts). The permission and automatic expiration time of the token can be defined as required upon issuing.

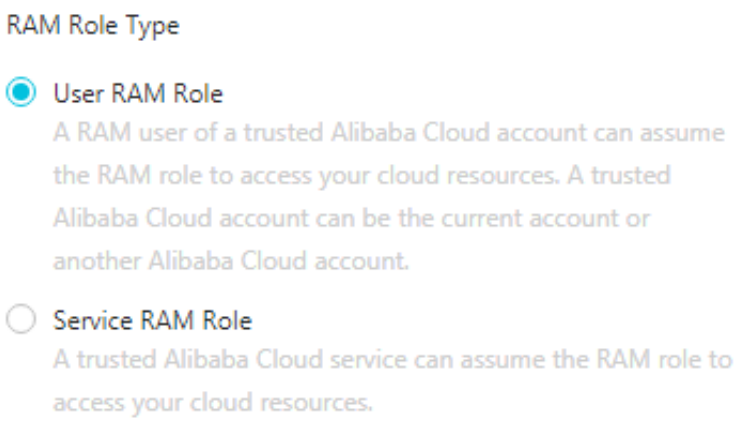
The advantage of using the STS access token to authorize temporary users is making the authorization more controllable. You do not need to create a RAM user account and key for the temporary users. The RAM user account and key are valid in the long term but the temporary users do not need to access the resources for long. For use cases, see *Grant temporary permissions to mobile apps* and *Cross-account resource authorization and access*.

Create a role

1. On the RAM console, choose RAM Roles > Create RAM Role

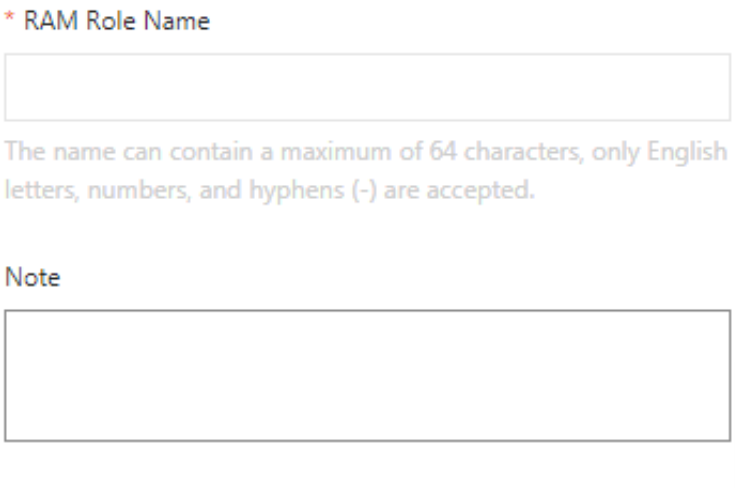


2. Select the role type. Here, the role User is selected.



3. Enter the type information. A subaccount of a trusted account can play the created role.

4. Enter the role name.



5. After a role is created, authorize the role. For details, see *Permission granting in RAM* and *Authorized resources*.

Temporary access authorization

Before using STS for access authorization, authorize the role to be assumed by the subaccount of the trusted cloud account created in Step 3. If any subaccount could assume these roles, unpredictable risks may occur. Therefore, in order to assume the corresponding role, a subaccount has to have explicitly configured permissions.

Authorization of the trusted cloud account

1. Click Policy Management on the left side of the page to go to the Policy Management page.
2. Click Create Authorization Policy on the right side of the page to go to the Create Authorization Policy page.
3. Select a blank template to go to the Create Custom Authorization Policy page.
4. Enter the authorization policy name and fill the following content to the policy content field.

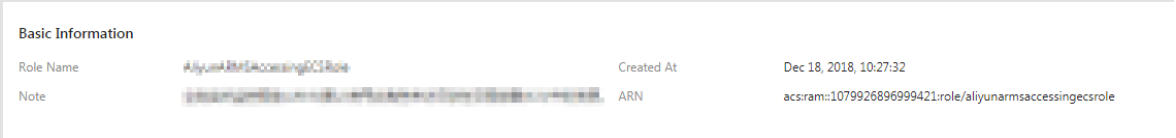
```
{
" Version ": " 1 ",
" Statement ": [
{
   " Effect ": " Allow ",
   " Action ": " sts : AssumeRole ",
   " Resource ": " acs : ram ::${ aliyunID }: role /${ roleName }"
}
]
```

```
}
```

${aliyunID} indicates the ID of the user that creates the role.

${roleName} indicates the role name in lowercase.

Note:
The resource details can be obtained from the Arn field in Role Details and Basic Information.



5. On the User Management page, authorize the permission of the role created for the subaccount. For details, see *Permission granting in RAM*.

Role assumed by a subaccount

After logging on to the console through the subaccount, the subaccount can switch to the authorized role assumed by the subaccount to practise permissions of the role. The steps are as follows:

1. Move the mouse to the profile picture on the upper-right corner of the navigation bar, and click Switch Role in the window.
2. Enter the enterprise alias of the account with which you intend to create a role. If the enterprise alias is not modified, the account ID is used by default. Enter the role name and then click Switch to switch to the specified role.

5 ElasticFlow

5.1 Source instance overview

ElasticFlow imports data from source instances, processes the data, and sends the data to downstream nodes. The source instance and data processing task configurations can be configured separately. You can first configure a source instance, and then select the source instance when configuring a task.

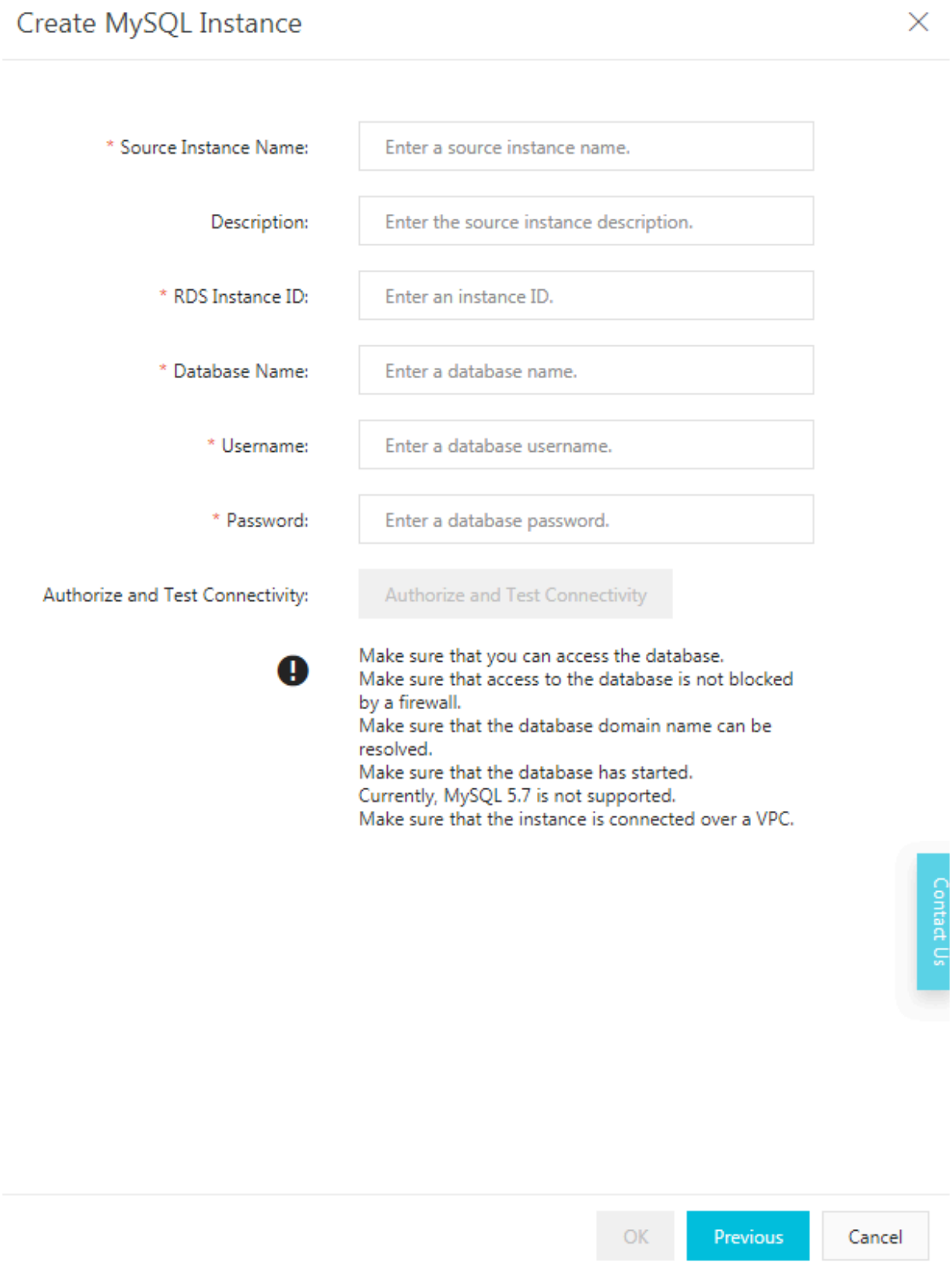
Source instance types

ElasticFlow supports the following Alibaba Cloud source instances. Some source instances require authorization. You can follow the instructions in the console to complete the authorization when creating a source instance.

- ApsaraDB RDS for MySQL

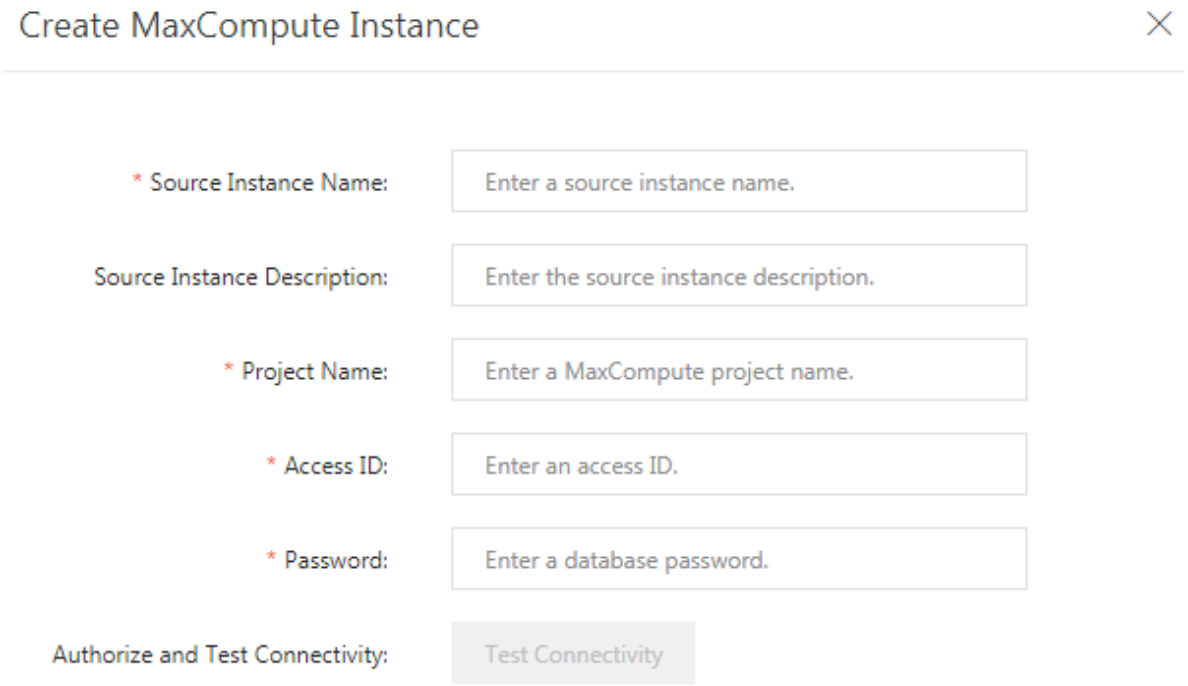
  To create an ApsaraDB RDS for MySQL source instance, add the prepared Alibaba Cloud RDS instance ID, database name, username, and password to ElasticFlow. The account must have the read permission. You can then select a table from

the source instance when creating a data import task. After the task is started, ElasticFlow will import data from the table on the source instance.



- MaxCompute

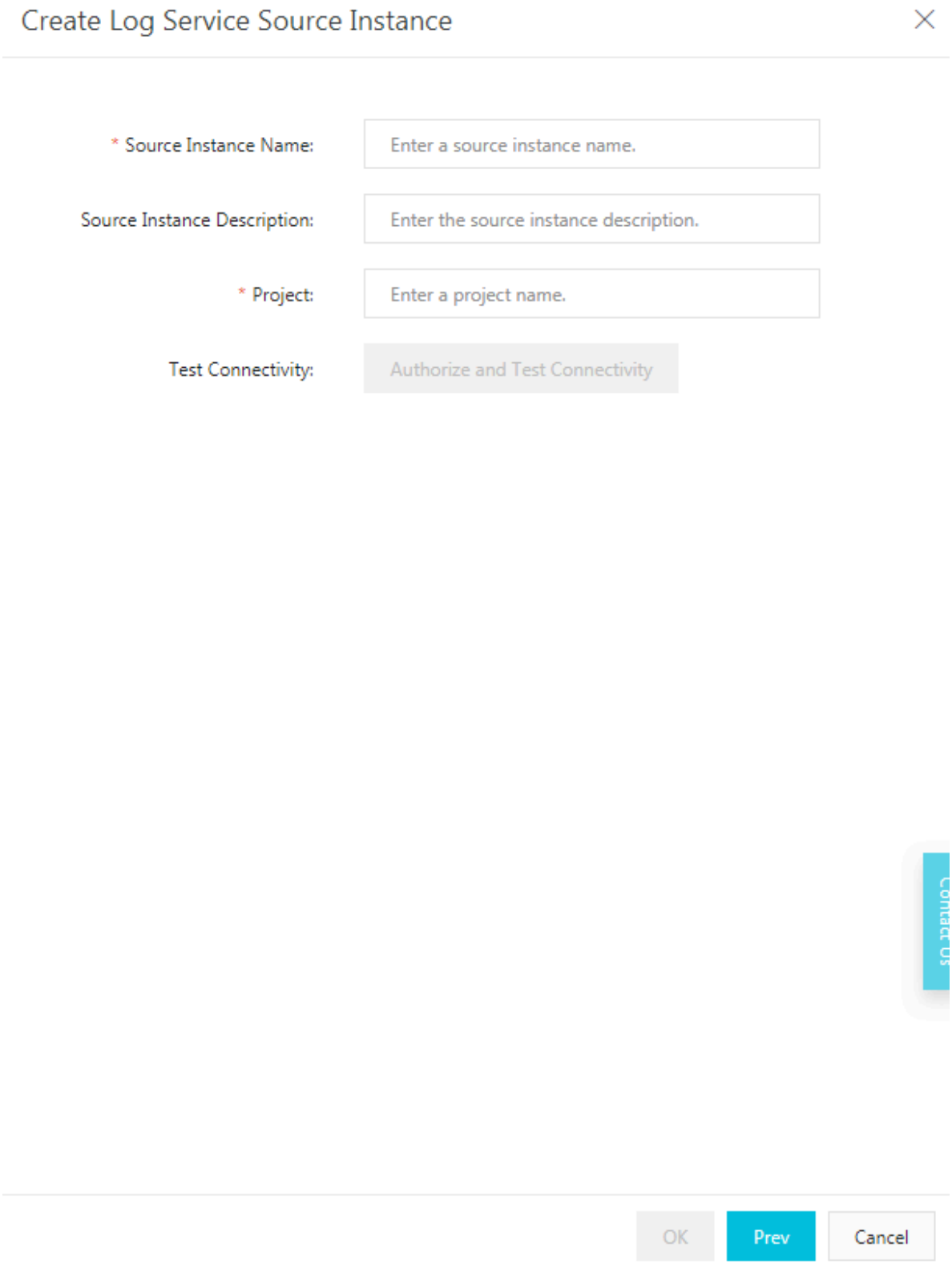
  To create a MaxCompute source instance, add the prepared Alibaba Cloud MaxCompute project name, AccessKey ID, and AccessKey secret to ElasticFlow . The account must have the read permission. You can then select a table from the source instance when creating a data import task. After the task is started, ElasticFlow will import data from the table on the source instance.



- Log Service

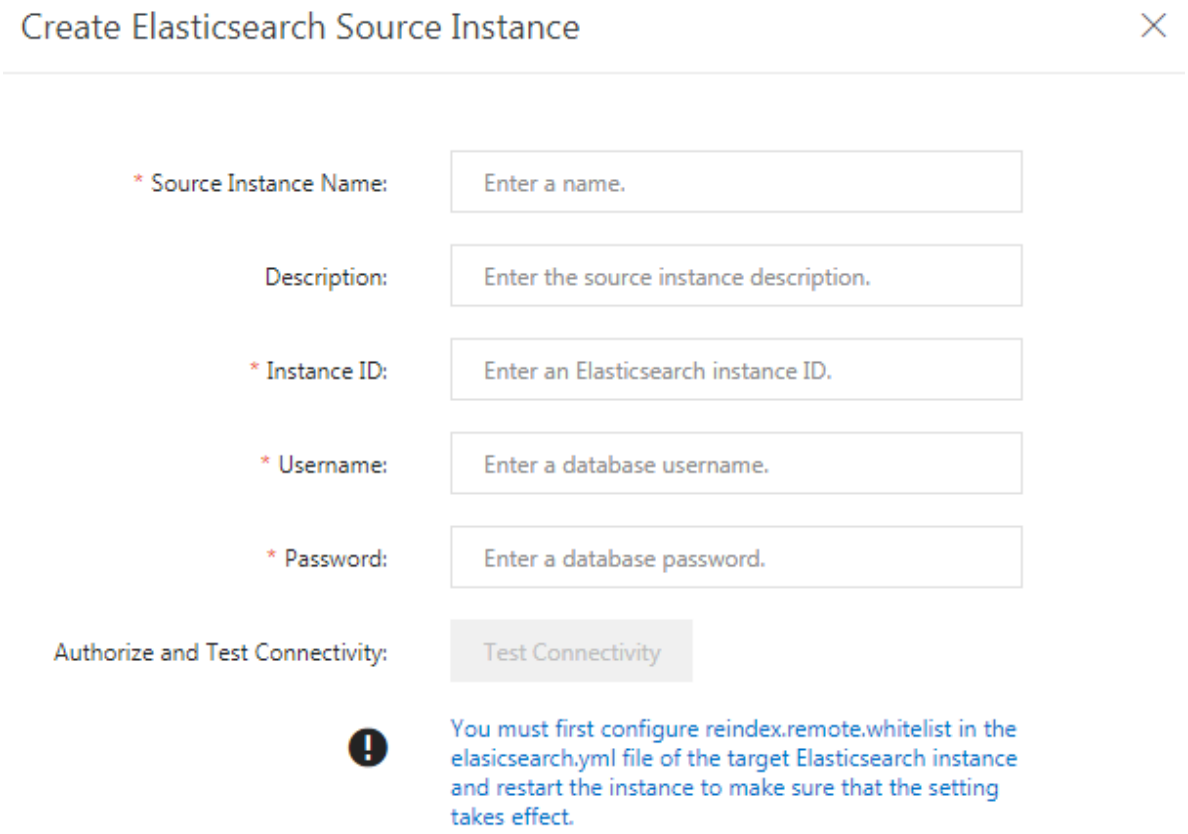
  To create a Log Service source instance, add the prepared Log Service project name to ElasticFlow. You can then select a table from the source instance when creating

a data import task. After the task is started, ElasticFlow will import data from the table on the source instance.



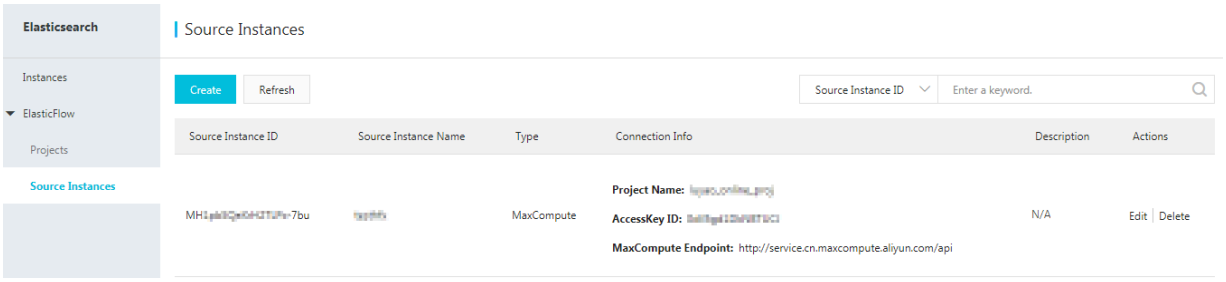
- Elasticsearch

  To create an Elasticsearch source instance, add the prepared Alibaba Cloud Elasticsearch instance ID, AccessKey ID, and AccessKey Secret to ElasticFlow. The account must have the read permission. You can then select a table from the source instance when creating a data import task. After the task is started, ElasticFlow will import data from the table on the source instance.



Source instance management

You can create, search, edit, and delete source instances on the source instances page. For more information about creating source instances, see *Create a source instance based on service authorization*.



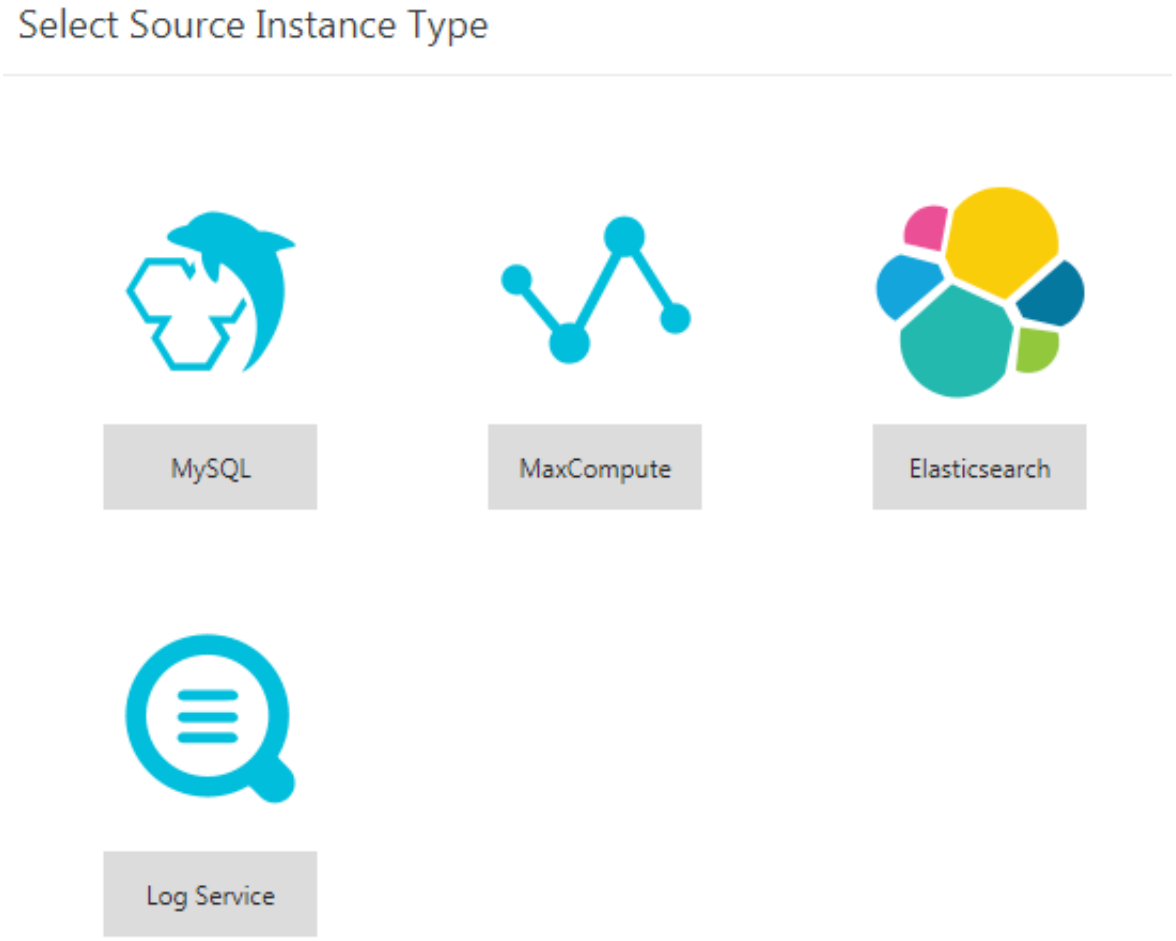
5.2 Quick start

5.2.1 Create a source instance based on service authorization

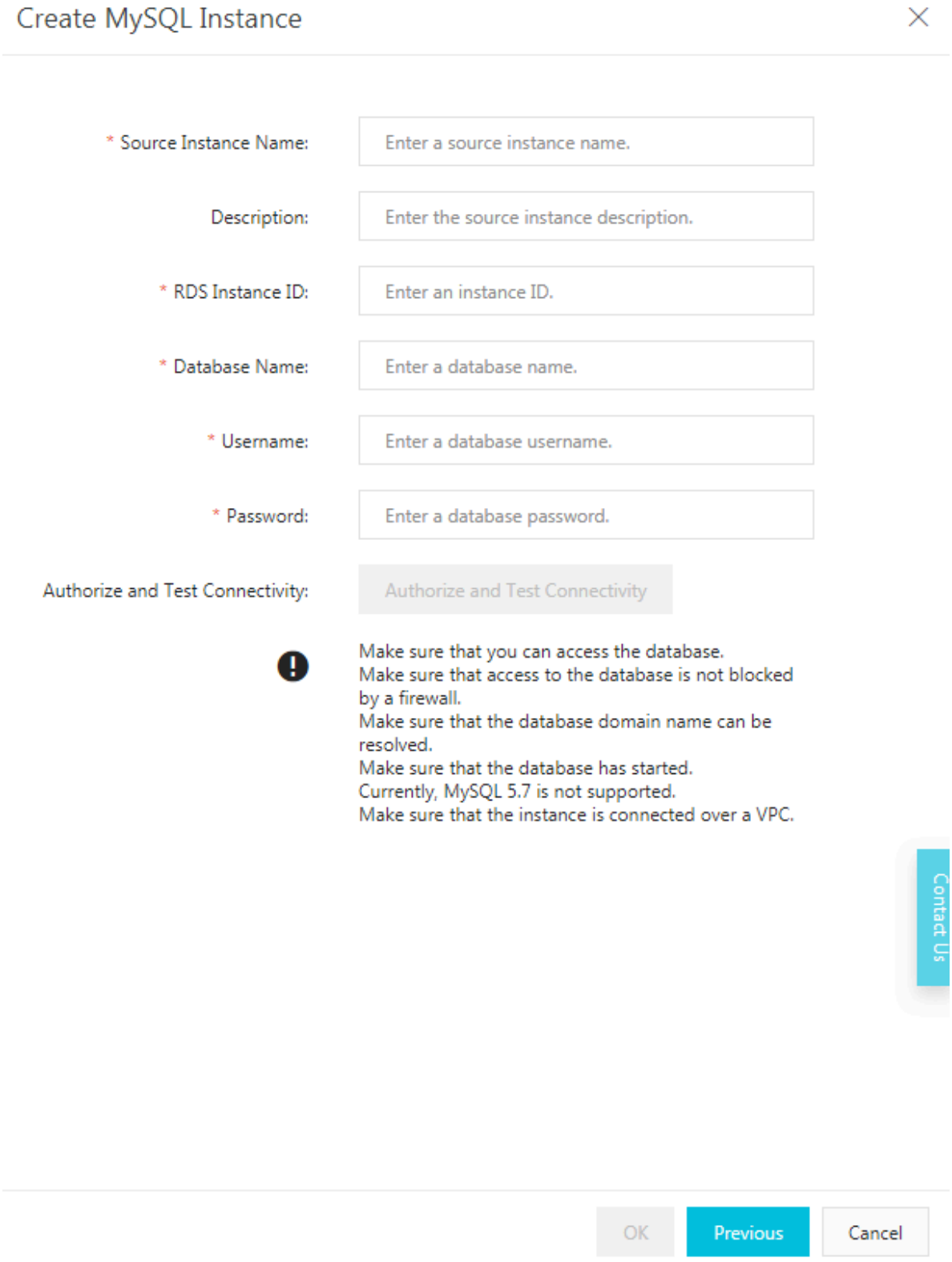
On the source instances page, you can create and manage source instances.

Create a source instance

1. Go to the ElasticFlow page.
2. In the left-side navigation pane, click Source Instances.
3. On the source instances page, click Create.
4. Select a source instance type, such as MySQL, and then click Next.

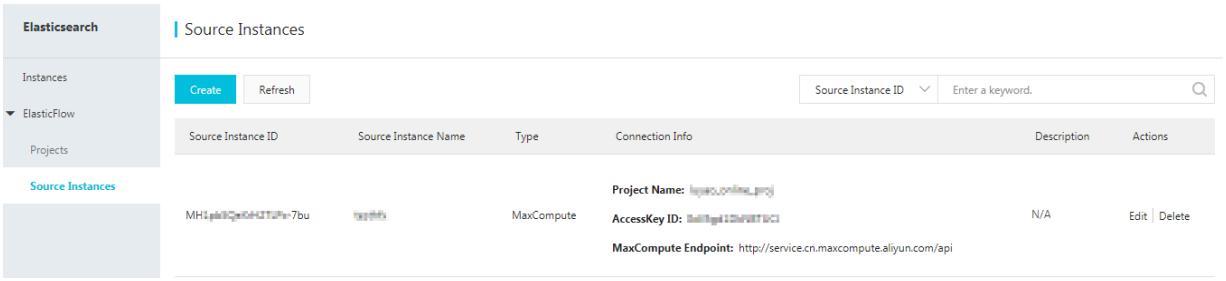


5. On the create source instance page, set parameters and click OK. The system will create the source instance after the configuration is validated.



Manage source instances

After creating a source instance, you can perform the following actions on the source instances page:



- Enter a source instance ID or name into the search box to search for the source instance.
- Create source instances.
- Edit source instances. Before you edit a source instance, make sure that no tasks are importing data from the source instance. Otherwise, stop the tasks first.
- Delete source instances. Before you delete a source instance, make sure that no tasks are importing data from the source instance. Otherwise, stop the tasks first.