Cloud Eye

API Reference

Issue 01

Date 2022-04-12

HUAWEI TECHNOLOGIES CO., LTD.

Trademarks and Permissions


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

Notice

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

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

Huawei Technologies Co., Ltd.

Address: Huawei Industrial Base
Bantian, Longgang
Shenzhen 518129
People's Republic of China

Website: https://www.huawei.com

Email: support@huawei.com

Contents

1 Before You Start

1.1 Overview

Welcome to *Cloud Eye API Reference*. Cloud Eye is a multi-dimensional resource monitoring platform. Customers can use Cloud Eye to monitor the utilization of service resources, track the running status of cloud services, configure alarm rules and notifications, and quickly respond to resource changes.

This document describes how to use application programming interfaces (APIs) to perform operations on metrics, alarm rules, and monitoring data, such as querying the metric list and the alarm rule list, creating alarm rules, and deleting alarm rules. For details about all supported operations, see **API Overview**.

If you plan to access Cloud Eye through an API, ensure that you are familiar with Cloud Eye concepts. For details, see "What Is Cloud Eye?" in the *Cloud Eye User Guide*.

1.2 API Calling

Cloud Eye supports Representational State Transfer (REST) APIs, allowing you to call APIs using HTTPS. For details about API calling, see **Calling APIs**.

1.3 Endpoints

An endpoint is the **request address** for calling an API. Endpoints vary depending on services and regions. For the endpoints of all services, see **Regions and Endpoints**.

1.4 Notes and Constraints

- The number of alarm rules that you can create is determined by your quota. To view or increase the quota, see "Quota Adjustment" in the *Cloud Eye User Guide*.
- For more constraints, see API description.

1.5 Concepts

- Account

  An account is created upon successful registration. The account has full access permissions for all of its cloud services and resources. It can be used to reset user passwords and grant user permissions. The account is a payment entity, which should not be used directly to perform routine management. For security purposes, create Identity and Access Management (IAM) users and grant them permissions for routine management.

- User

  An IAM user is created by an account in IAM to use cloud services. Each IAM user has its own identity credentials (password and access keys).

  API authentication requires information such as the account name, username, and password.

- Region

  A region is a geographic area in which cloud resources are deployed. Availability zones (AZs) in the same region can communicate with each other over an intranet, while AZs in different regions are isolated from each other. Deploying cloud resources in different regions can better suit certain user requirements or comply with local laws or regulations.

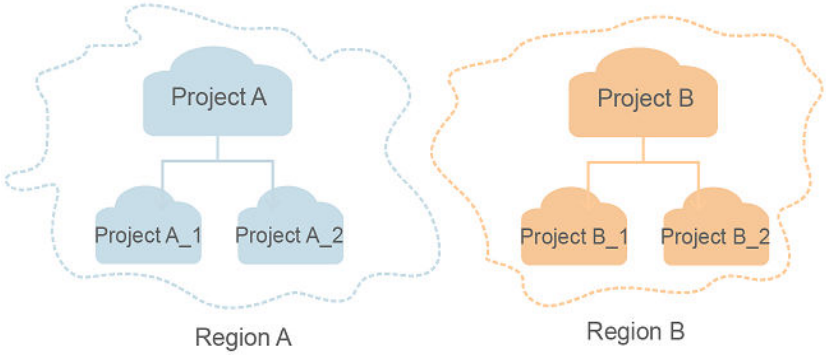
- AZ

  An AZ comprises of one or more physical data centers equipped with independent ventilation, fire, water, and electricity facilities. Computing, network, storage, and other resources in an AZ are logically divided into multiple clusters. AZs within a region are interconnected using high-speed optical fibers to allow you to build cross-AZ high-availability systems.

- Project

  A project corresponds to a region. Default projects are defined to group and physically isolate resources (including computing, storage, and network resources) across regions. Users can be granted permissions in a default project to access all resources under their accounts in the region associated with the project. If you need more refined access control, create subprojects under a default project and create resources in subprojects. Then you can assign users the permissions required to access only the resources in the specific subprojects.

  Figure 1-1 Project isolation model



- Enterprise project

  Enterprise projects group and manage resources across regions. Resources in different enterprise projects are logically isolated. An enterprise project can contain resources of multiple regions, and resources can be added to or removed from enterprise projects.

  For details about enterprise projects and about how to obtain enterprise project IDs, see *Enterprise Management User Guide*.

2 API Overview

Cloud Eye APIs allow you to use all Cloud Eye functions. For example, you can query the metric list and create alarm rules.

Table 2-1 API description

Type	Subtype	API	Description
Cloud Eye API	API version management	**Querying All API Versions**	Query all API versions supported by Cloud Eye.
		Querying a Specified API Version	Query a specified API version supported by Cloud Eye.
	Metric management	**Querying Metrics**	Query the list of metrics that currently monitored by Cloud Eye.
	Alarm rule management	**Querying Alarm Rules**	Query the alarm rule list.
		Querying an Alarm Rule	Query the alarm rule information based on the alarm rule ID.
		Enabling or Disabling an Alarm Rule	Enable or disable an alarm rule based on the alarm rule ID.
		Deleting an Alarm Rule	Delete an alarm rule based on the alarm rule ID.
		Creating an Alarm Rule	Create an alarm rule.
	Monitoring data management	**Querying Monitoring Data**	Query the monitoring data of a specified metric of specified granularity in a specified time range.

Type	Subtype	API	Description
		Adding Monitoring Data	Add one or more pieces of metric monitoring data.
	Quota management	**Querying Quotas**	Query the alarm rule quota.
	Event monitoring	**Reporting Events**	Report custom events.

3 Calling APIs

3.1 Making an API Request

This section describes the structure of a REST API request, and uses the IAM API for **obtaining a user token** as an example to demonstrate how to call an API. The obtained token can then be used to authenticate the calling of other APIs.

Request URI

A request URI is in the following format:

{URI-scheme}://{Endpoint}/{resource-path}?{query-string}

Although a request URI is included in the request header, most programming languages or frameworks require the request URI to be transmitted separately.

Table 3-1 URI parameter description

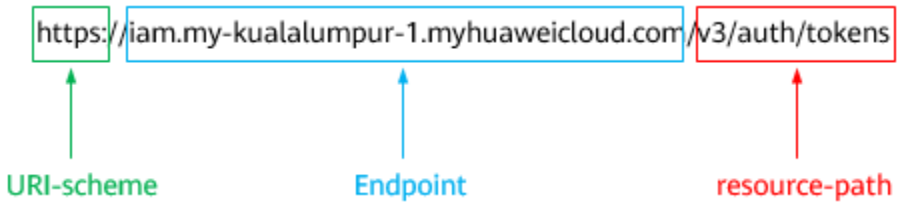
Parameter	Description
URI-scheme	Protocol used to transmit requests. All APIs use HTTPS.
Endpoint	Domain name or IP address of the server bearing the REST service. The endpoint varies between services in different regions. It can be obtained from **Regions and Endpoints**. For example, the endpoint of IAM in the **my-kualalumpur-1** region is **iam.my-kualalumpur-1.myhuaweicloud.com**.
resource-path	Access path of an API for performing a specified operation. Obtain the path from the URI of an API. For example, the **resource-path** of the API used to obtain a user token is **/v3/auth/tokens**.

Parameter	Description
query-string	Query parameter, which is optional. Ensure that a question mark (?) is included before each query parameter that is in the format of *Parameter name*=*Parameter value*. For example, **?limit=10** indicates that a maximum of 10 data records will be displayed.

For example, to obtain an IAM token in the **AP-Kuala Lumpur-OP6** region, obtain the endpoint of IAM (**iam.my-kualalumpur-1.myhuaweicloud.com**) for this region and the **resource-path** (**/v3/auth/tokens**) in the URI of the API used to **obtain a user token**. Then, construct the URI as follows:

```
https://iam.my-kualalumpur-1.myhuaweicloud.com/v3/auth/tokens
```

Figure 3-1 Example URI



NOTE

To simplify the URI display in this document, each API is provided only with a **resource-path** and a request method. The **URI-scheme** of all APIs is **HTTPS**, and the endpoints of all APIs in the same region are identical.

Request Methods

The HTTP protocol defines the following request methods that can be used to send a request to the server.

Table 3-2 HTTP methods

Method	Description
GET	Requests the server to return specified resources.
PUT	Requests the server to update specified resources.
POST	Requests the server to add resources or perform special operations.
DELETE	Requests the server to delete specified resources, for example, an object.
HEAD	Same as GET except that the server must return only the response header.

Method	Description
PATCH	Requests the server to update partial content of a specified resource. If the resource does not exist, a new resource will be created.

For example, in the case of the API used to **obtain a user token**, the request method is **POST**. The request is as follows:

```
POST https://iam.my-kualalumpur-1.myhuaweicloud.com/v3/auth/tokens
```

Request Header

You can also add additional header fields to a request, such as the fields required by a specified URI or HTTP method. For example, to request for the authentication information, add **Content-Type**, which specifies the request body type.

Common request header fields are as follows.

Table 3-3 Common request header fields

Parameter	Description	Mandatory	Example Value
Host	Specifies the server domain name and port number of the resources being requested. The value can be obtained from the URL of the service API. The value is in the format of *Hostname:Port number*. If the port number is not specified, the default port is used. The default port number for **https** is **443**.	No This field is mandatory for AK/SK authentication.	code.test.com or code.test.com:443
Content-Type	Specifies the type (or format) of the message body. The default value **application/json** is recommended. Other values of this field will be provided for specific APIs if any.	Yes	application/json
Content-Length	Specifies the length of the request body. The unit is byte.	No	3495

Parameter	Description	Mandatory	Example Value
X-Project-Id	Specifies the project ID. Obtain the project ID by following the instructions in **Obtaining a Project ID**.	No	e9993fc787d94b6c886cbaa340f9c0f4
X-Auth-Token	Specifies the user token. It is a response to the API for **obtaining a user token** (This is the only API that does not require authentication). After the request is processed, the value of **X-Subject-Token** in the response header is the token value.	No This field is mandatory for token authentication.	The following is part of an example token: MIIPAgYJKoZIhvcNAQcCo...ggg1BBIINPXsidG9rZ

NOTE

In addition to supporting authentication using tokens, APIs support authentication using AK/SK, which uses SDKs to sign a request. During the signature, the **Authorization** (signature authentication) and **X-Sdk-Date** (time when a request is sent) headers are automatically added in the request.

For more details, see "Authentication Using AK/SK" in **Authentication**.

The API used to **obtain a user token** does not require authentication. Therefore, only the **Content-Type** field needs to be added to requests for calling the API. An example of such requests is as follows:

```
POST https://iam.my-kualalumpur-1.myhuaweicloud.com/v3/auth/tokens
Content-Type: application/json
```

(Optional) Request Body

This part is optional. The body of a request is often sent in a structured format as specified in the **Content-Type** header field. The request body transfers content except the request header.

The request body varies between APIs. Some APIs do not require the request body, such as the APIs requested using the GET and DELETE methods.

In the case of the API used to **obtain a user token**, the request parameters and parameter description can be obtained from the API request. The following provides an example request with a body included. Replace *username*, *domainname*, ******** (login password), and *xxxxxxxxxxxxxxxxx* (project name) with the actual values. Obtain a project name from **Regions and Endpoints**.

> **NOTE**
>
> The **scope** parameter specifies where a token takes effect. You can set **scope** to an account or a project under an account. In the following example, the token takes effect only for the resources in a specified project. For more information about this API, see **Obtaining a User Token**.

```
POST https://iam.my-kualalumpur-1.myhuaweicloud.com/v3/auth/tokens
Content-Type: application/json

{
    "auth": {
        "identity": {
            "methods": [
                "password"
            ],
            "password": {
                "user": {
                    "name": "username",
                    "password": "********",
                    "domain": {
                        "name": "domainname"
                    }
                }
            }
        },
        "scope": {
            "project": {
                "name": "xxxxxxxxxxxxxxxxx"
            }
        }
    }
}
```

If all data required for the API request is available, you can send the request to call the API through **curl**, **Postman**, or coding. In the response to the API used to obtain a user token, **x-subject-token** is the desired user token. This token can then be used to authenticate the calling of other APIs.

3.2 Authentication

Requests for calling an API can be authenticated using either of the following methods:

- Token authentication: Requests are authenticated using tokens.
- AK/SK authentication: Requests are encrypted using AK/SK pairs. AK/SK authentication is recommended because it is more secure than token authentication.

Token Authentication

> **NOTE**
>
> The validity period of a token is 24 hours. When using a token for authentication, cache it to prevent frequently calling the IAM API used to obtain a user token.

A token specifies temporary permissions in a computer system. During API authentication using a token, the token is added to requests to get permissions for calling the API. You can obtain a token by calling the **Obtaining User Token** API.

A cloud service can be deployed as either a project-level service or global service.

- For a project-level service, you need to obtain a project-level token. When you call the API, set **auth.scope** in the request body to **project**.
- For a global service, you need to obtain a global token. When you call the API, set **auth.scope** in the request body to **domain**.

IMS is a project-level service. When you call the API, set **auth.scope** in the request body to **project**.

```
{
    "auth": {
        "identity": {
            "methods": [
                "password"
            ],
            "password": {
                "user": {
                    "name": "username",
                    "password": "********",
                    "domain": {
                        "name": "domainname"
                    }
                }
            }
        },
        "scope": {
            "project": {
                "name": "xxxxxxxx"
            }
        }
    }
}
```

After a token is obtained, the **X-Auth-Token** header field must be added to requests to specify the token when calling other APIs. For example, if the token is **ABCDEFJ....**, **X-Auth-Token: ABCDEFJ....** can be added to a request as follows:

```
POST https://iam.my-kualalumpur-1.myhuaweicloud.com/v3/auth/projects
Content-Type: application/json
X-Auth-Token: ABCDEFJ....
```

AK/SK Authentication

NOTE

AK/SK authentication supports API requests with a body not larger than 12 MB. For API requests with a larger body, token authentication is recommended.

In AK/SK authentication, AK/SK is used to sign requests and the signature is then added to the requests for authentication.

- AK: access key ID, which is a unique identifier used in conjunction with a secret access key to sign requests cryptographically.
- SK: secret access key, which is used in conjunction with an AK to sign requests cryptographically. It identifies a request sender and prevents the request from being modified.

In AK/SK authentication, you can use an AK/SK to sign requests based on the signature algorithm or using the signing SDK. For details about how to sign requests and use the signing SDK, see .

NOTE

The signing SDK is only used for signing requests and is different from the SDKs provided by services.

3.3 Response

Status Code

After sending a request, you will receive a response, including a status code, response header, and response body.

A status code is a group of digits, ranging from 1xx to 5xx. It indicates the status of a request. For more information, see **Status Codes**.

For example, if status code **201** is returned for calling the API used to **obtain a user token**, the request is successful.

Response Header

Similar to a request, a response also has a header, for example, **Content-Type**.

Figure 3-2 shows the response header fields for the API used to **obtain a user token**. The **x-subject-token** header field is the desired user token. This token can then be used to authenticate the calling of other APIs.

Figure 3-2 Header fields of the response to the request for obtaining a user token

(Optional) Response Body

The body of a response is often returned in structured format as specified in the **Content-Type** header field. The response body transfers content except the response header.

The following is part of the response body for the API used to **obtain a user token**.

```
{
    "token": {
        "expires_at": "2019-02-13T06:52:13.855000Z",
        "methods": [
            "password"
        ],
        "catalog": [
            {
                "endpoints": [
                    {
                        "region_id": "az-01",
.....
```

If an error occurs during API calling, an error code and a message will be displayed. The following shows an error response body.

```
{
    "error_msg": "The format of message is error",
    "error_code": "AS.0001"
}
```

In the response body, **error_code** is an error code, and **error_msg** provides information about the error.

4 Getting Started

Overview

This topic describes how to invoke a number of Cloud Eye APIs to create an alarm rule for the ECS CPU usage.

NOTE

The validity period of a token obtained from IAM is 24 hours. If you want to use a token for authentication, cache it to avoid frequently calling the IAM API.

Creation Procedure

1. **Obtain the user token.**
2. **Query the list of metrics that can be monitored.**
3. **Create an alarm rule.**

Procedure

1. Obtain the user token.

   Send **POST https://***IAM endpoint***/v3/auth/tokens**.

   Add **Content-Type:application/json** to the request headers.

   The request body is as follows:

```
{
  "auth": {
    "identity": {
      "methods": [
        "password"
      ],
      "password": {
        "user": {
          "name": "James",
          "password": "**********",
          "domain": {
            "name": "A-Company"
          }
        }
      }
    },
    "scope": {
      "project": {
        "name": "XXX",
```



```
      "domain": {
        "name": "A-Company"
      }
    }
  }
 }
}
```

Specify the following parameters:

- **user.name**: username, which is set based on the obtained token body
- **password**: login password
- **domain.name:**: name of the account to which the user belongs. If the account is used to obtain the token, values of **user.name** of the account and **domain.name** are the same. In this case, enter the **user.name** value. Otherwise, enter the domain name to which the account belongs.
- **project.name**: region

  NOTE

  Obtain **X-Subject-Token** from the response header, that is, the signed token.

2. Query the list of metrics that can be monitored.

   Send **GET https://***Cloud Eye endpoint***/V1.0/{project_id}/metrics**.

   Add **X-Auth-Token** obtained in **1** to the request header.

   After the request is successfully responded, the **metrics** information is returned, such as **"metric_name": "cpu_util"** in the following figure.

```
{
  "metrics": [
    {
      "namespace": "SYS.ECS",
      "dimensions": [
        {
          "name": "instance_id",
          "value": "d9112af5-6913-4f3b-bd0a-3f96711e004d"
        }
      ],
      "metric_name": "cpu_util",
      "unit": "%"
    }
  ],
  "meta_data": {
    "count": 1,
    "marker": "SYS.ECS.cpu_util.instance_id:d9112af5-6913-4f3b-bd0a-3f96711e004d",
    "total": 7
  }
}
```

   If the request fails, an error code and error information are returned. For details, see **Error Codes**.

3. Create an alarm rule.

   Send **POST https://***Cloud Eye endpoint***/V1.0/{project_id}/alarms**.

   Specify the following parameters in the request body:

```
{
  "alarm_name": "alarm-rp0E",  //Alarm rule name (mandatory, string)
  "alarm_description": "",
  "metric": {
    "namespace": "SYS.ECS",  //Namespace (mandatory, string)
    "dimensions": [
      {
        "name": "instance_id",
        "value": "33328f02-3814-422e-b688-bfdba93d4051"
```



```
            }
        ],
        "metric_name": "cpu_util"   //Metric name (mandatory, string)
    },
    "condition": {
        "period": 300,      //Monitoring period (mandatory, integer)
        "filter": "average",     //Data rollup method (mandatory, string)
        "comparison_operator": ">=",    //Operator of the alarm threshold (mandatory, string)
        "value": 80, //Threshold (mandatory, string)
        "unit": "%",  //Data unit (mandatory, string)
        "count": 1
    },
    "alarm_enabled": true,
    "alarm_action_enabled": true,
    "alarm_level": 2,
    "alarm_actions": [
        {
            "type": "notification",
            "notificationList": [ ]
        }
    ],
    "ok_actions": [
        {
            "type": "notification",
            "notificationList": [ ]
        }
    ]
}
```

If the request is responded, the alarm rule ID is returned.

```
{
    "alarm_id":"al1450321795427dR8p5mQBo"
}
```

If the request fails, an error code and error information are returned. For details, see **Error Codes**.

You can query, enable, disable, or delete alarm rules based on the alarm rule ID obtained in **3**.

5 API Description

5.1 API Version Management

5.1.1 Querying All API Versions

Function

This API is used to query all API versions supported by Cloud Eye.

URI

GET /

Request

Example request

```
GET https://{Cloud Eye endpoint}/
```

Response

- Response parameters

Table 5-1 Parameter description

Parameter	Type	Description
versions	Array of objects	Specifies the list of all versions. For details, see **Table 5-2**.

Table 5-2 versions data structure description

Parameter	Type	Description
id	String	Specifies the version ID, for example, v1.
links	Array of objects	Specifies the API URL. For details, see **Table 5-3**.
version	String	Specifies the API version. If the APIs of this version support microversions, set this parameter to the supported maximum microversion. If the microversion is not supported, leave this parameter blank.
status	String	Specifies the version status. **CURRENT**: indicates a primary version. **SUPPORTED**: indicates an old version but is still supported. **DEPRECATED**: indicates a deprecated version which may be deleted later.
updated	String	Specifies the version release time, which must be the UTC time. For example, the release time of v1 is **2014-06-28T12:20:21Z**.
min_version	String	If the APIs of this version support microversions, set this parameter to the supported minimum microversion. If not, leave this parameter blank.

Table 5-3 links data structure description

Parameter	Type	Description
href	String	Specifies the reference address of the current API version.
rel	String	Specifies the relationship between the current API version and the referenced address.

- Example response

```
{
 "versions": [
  {
   "id": "V1.0",
   "links": [
    {
     "href": "https://x.x.x.x/V1.0/",
     "rel": "self"
    }
   ],
   "min_version": "",
   "status": "CURRENT",
   "updated": "2018-09-30T00:00:00Z",
   "version": ""
  }
```



```
 ]
}
```

Returned Values

- Normal

  200

- Abnormal

Returned Value	Description
400 Bad Request	Request error.
401 Unauthorized	The authentication information is not provided or is incorrect.
403 Forbidden	Access to the requested page is forbidden.
408 Request Timeout	The request timed out.
429 Too Many Requests	Concurrent requests are excessive.
500 Internal Server Error	Failed to complete the request because of an internal service error.
503 Service Unavailable	The service is currently unavailable.

Error Codes

See **Error Codes**.

5.1.2 Querying a Specified API Version

Function

This API is used to query a specified API version of Cloud Eye.

URI

GET /{api_version}

- Parameter description

  Table 5-4 Parameter description

Parameter	Mandatory	Description
api_version	Yes	Specifies the API version.

- Example

```
GET https://{Cloud Eye endpoint}/V1.0\
```


Request

None

Response

- Response parameters

Table 5-5 Parameter description

Parameter	Type	Description
version	Objects	Specifies the list of all versions. For details, see **Table 5-6**.

Table 5-6 versions data structure description

Parameter	Type	Description
id	String	Specifies the version ID, for example, v1.
links	Array of objects	Specifies the API URL. For details, see **Table 5-7**.
version	String	Specifies the API version. If the APIs of this version support microversions, set this parameter to the supported maximum microversion. If the microversion is not supported, leave this parameter blank.
status	String	Specifies the version status. **CURRENT**: indicates a primary version. **SUPPORTED**: indicates an old version but is still supported. **DEPRECATED**: indicates a deprecated version which may be deleted later.
updated	String	Specifies the version release time, which must be the UTC time. For example, the release time of v1 is **2014-06-28T12:20:21Z**.
min_version	String	If the APIs of this version support microversions, set this parameter to the supported minimum microversion. If not, leave this parameter blank.

Table 5-7 links data structure description

Parameter	Type	Description
href	String	Specifies the reference address of the current API version.
rel	String	Specifies the relationship between the current API version and the referenced address.

- Example response

```
{
  "version": {
     "id": "V1.0",
     "links": [
       {
         "href": "https://x.x.x.x/V1.0/",
         "rel": "self"
       }
     ],
     "min_version": "",
     "status": "CURRENT",
     "updated": "2018-09-30T00:00:00Z",
     "version": ""
   }
}
```

Returned Values

- Normal

  200

- Abnormal

Returned Value	Description
400 Bad Request	Request error.
401 Unauthorized	The authentication information is not provided or is incorrect.
403 Forbidden	Access to the requested page is forbidden.
408 Request Timeout	The request timed out.
429 Too Many Requests	Concurrent requests are excessive.
500 Internal Server Error	Failed to complete the request because of an internal service error.
503 Service Unavailable	The service is currently unavailable.

Error Codes

See **Error Codes**.

5.2 Metric Management

5.2.1 Querying Metrics

Function

This API is used to query the metrics. You can specify the namespace, metric, dimension, sorting order, start records, and the maximum number of records when using this API to query metrics.

URI

GET /V1.0/{project_id}/metrics

- Parameter description

Table 5-8 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.

Table 5-9 Query parameter description

Parameter	Mandatory	Type	Description
namespace	No	String	Query the namespace of a service. For details, see **Services Interconnected with Cloud Eye**. The namespace must be in the **service.item** format and contain 3 to 32 characters. **service** and **item** each must start with a letter and contain only letters, digits, and underscores (_).
metric_name	No	String	Specifies the metric ID. For example, if the monitoring metric of an ECS is CPU usage, **metric_name** is **cpu_util**. For details, see **Services Interconnected with Cloud Eye**.

Parameter	Mandatory	Type	Description
dim	No	String	Specifies the dimension. For example, the ECS dimension is **instance_id**. For details about each service dimension, see **Services Interconnected with Cloud Eye**. A maximum of three dimensions are supported, and the dimensions are numbered from 0 in **dim.{i}=key,value** format. **key** cannot exceed 32 characters and **value** cannot exceed 256 characters. Single dimension: **dim.0=instance_id,6f3c6f91-4b24-4e1b-b7d1-a94ac1cb011d** Multiple dimensions: **dim.0=key,value&dim.1=key,value**
start	No	String	Specifies the paging start value. The format is **namespace.metric_name.key:value**. Example: **start=SYS.ECS.cpu_util.instance_id:d9112af5-6913-4f3b-bd0a-3f96711e004d**.
limit	No	Integer	Supported range: **1** to **1000** (default) This parameter is used to limit the number of query results.
order	No	String	Specifies the result sorting method, which is sorted by timestamp. The default method is **desc**. • **asc**: The query results are displayed in the ascending order. • **desc**: The query results are displayed in the descending order.

- Example requests

  Example request 1: Query all metrics that can be monitored.

```
GET https://{Cloud Eye endpoint}/V1.0/{project_id}/metrics
```


Example request 2: Query the CPU usage of the ECS whose ID is **6f3c6f91-4b24-4e1b-b7d1-a94ac1cb011d**. Retain 10 records in descending order by timestamp.

```
GET https://{Cloud Eye endpoint}/V1.0/{project_id}/metrics?
namespace=SYS.ECS&metric_name=cpu_util&dim.0=instance_id,6f3c6f91-4b24-4e1b-b7d1-
a94ac1cb011d&limit=10&order=desc
```

Request

None

Response

- Response parameters

Table 5-10 Parameter description

Parameter	Type	Description
metrics	Array of objects	Specifies the list of metric objects. For details, see **Table 5-11**.
meta_data	Object	Specifies the metadata of query results, including the pagination information. For details, see **Table 5-13**.

Table 5-11 metrics data structure description

Parameter	Type	Description
namespace	String	Specifies the metric namespace.
dimensions	Array of objects	Specifies the list of metric dimensions. For details, see **Table 5-12**.
metric_name	String	Specifies the metric name, such as **cpu_util**.
unit	String	Specifies the metric unit.

Table 5-12 dimensions data structure description

Parameter	Type	Description
name	String	Specifies the dimension. For example, the ECS dimension is **instance_id**. For details about the dimension of each service, see the **key** column in **Services Interconnected with Cloud Eye**.

Parameter	Type	Description
value	String	Specifies the dimension value, for example, an ECS ID. Enter 1 to 256 characters.

Table 5-13 meta_data data structure description

Parameter	Type	Description
count	Integer	Specifies the number of returned results.
marker	String	Specifies the pagination marker. For example, you have queried 10 records this time and the tenth record is about **cpu_util**. In your next query, if **start** is set to **cpu_util**, you can start your query from the next metric of **cpu_util**.
total	Integer	Specifies the total number of metrics.

- Example response

```
{
   "metrics": [
      {
         "namespace": "SYS.ECS",
         "dimensions": [
            {
               "name": "instance_id",
               "value": "d9112af5-6913-4f3b-bd0a-3f96711e004d"
            }
         ],
         "metric_name": "cpu_util",
         "unit": "%"
      }
   ],
   "meta_data": {
      "count": 1,
      "marker": "SYS.ECS.cpu_util.instance_id:d9112af5-6913-4f3b-bd0a-3f96711e004d",
      "total": 7
   }
}
```

Returned Values

- Normal

  200

- Abnormal

Returned Value	Description
400 Bad Request	Request error.
401 Unauthorized	The authentication information is not provided or is incorrect.

Returned Value	Description
403 Forbidden	Access to the requested page is forbidden.
408 Request Timeout	The request timed out.
429 Too Many Requests	Concurrent requests are excessive.
500 Internal Server Error	Failed to complete the request because of an internal service error.
503 Service Unavailable	The service is currently unavailable.

Error Codes

See **Error Codes**.

5.3 Alarm Rule Management

5.3.1 Querying Alarm Rules

Function

This API is used to query the alarm rule list. You can specify the paging parameters to limit the number of query results displayed on a page. You can also set the sorting order of query results.

URI

GET /V1.0/{project_id}/alarms

- Parameter description

Table 5-14 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.

Table 5-15 Parameter description

Parameter	Type	Description
alarms	Array of objects	Specifies the alarm rule list. For details, see **Table 5-16**.

Table 5-16 Query parameter description

Parameter	Mandatory	Type	Description
start	No	String	Specifies the first queried alarm to be displayed on a page. The value is **alarm_id**.
limit	No	Integer	Supported range: **1** to **100** (default) This parameter is used to limit the number of query results.
order	No	String	Specifies the result sorting method, which is sorted by timestamp. The default method is **desc**. • **asc**: The query results are displayed in the ascending order. • **desc**: The query results are displayed in the descending order.

- Example

  Request example 1: Query the current alarm rule list.

  ```
  GET https://{Cloud Eye endpoint}/V1.0/{project_id}/alarms
  ```

  Request example 2: Query the alarm rule list. Start by setting **alarm_id** to **al1441967036681YkazZ0deN** and retain 10 records in the descending order of time stamps.

  ```
  GET https://{Cloud Eye endpoint}/V1.0/{project_id}/alarms?
  start=al1441967036681YkazZ0deN&limit=10&order=desc
  ```

Request

None

Response

- Response parameters

Table 5-17 Response parameters

Parameter	Type	Description
metric_alarms	Array of objects	Specifies the list of alarm objects. For details, see **Table 5-18**.
meta_data	Object	Specifies the metadata of query results, including the pagination information. For details, see **Table 5-24**.

Table 5-18 metric_alarms data structure description

Parameter	Type	Description
alarm_name	String	Specifies the alarm rule name.
alarm_description	String	Provides supplementary information about the alarm rule.
metric	Object	Specifies the alarm metric. For details, see **Table 5-19**.
condition	Object	Specifies the alarm triggering condition. For details, see **Table 5-23**.
alarm_enabled	Boolean	Specifies whether to enable the alarm rule.
alarm_level	Integer	Specifies the alarm severity, which can be **1**, **2** (default), **3** or **4**, indicating critical, major, minor, and informational, respectively.
alarm_action_enabled	Boolean	Specifies whether to enable the action to be triggered by an alarm.
alarm_actions	Array of objects	Specifies the action to be triggered by an alarm. For details, see **Table 5-21**.
ok_actions	Array of objects	Specifies the action to be triggered after the alarm is cleared. For details, see **Table 5-22**.
alarm_id	String	Specifies the alarm rule ID.
update_time	Long	Specifies when the alarm status changed. The time is a UNIX timestamp and the unit is ms.

Parameter	Type	Description
alarm_state	String	Specifies the alarm status, which can be • **ok**: The alarm status is normal. • **alarm**: An alarm is generated. • **insufficient_data**: The required data is insufficient.

Table 5-19 metric data structure description

Parameter	Type	Description
namespace	String	Query the namespace of a service. For details, see **Services Interconnected with Cloud Eye**.
dimensions	Array of objects	Specifies the list of metric dimensions. For details, see **Table 5-20**.
metric_name	String	Specifies the metric ID. For example, if the monitoring metric of an ECS is CPU usage, **metric_name** is **cpu_util**. For details, see **Services Interconnected with Cloud Eye**.

Table 5-20 dimensions data structure description

Parameter	Type	Description
name	String	Specifies the dimension. For example, the ECS dimension is **instance_id**. For details about the dimension of each service, see the **key** column in **Services Interconnected with Cloud Eye**.
value	String	Specifies the dimension value, for example, an ECS ID. Enter 1 to 256 characters.

Table 5-21 alarm_actions data structure description

Parameter	Type	Description
type	String	Specifies the alarm notification type. • **notification**: indicates that a notification will be sent. • **autoscaling**: indicates that a scaling action will be triggered.

Parameter	Type	Description
notificationList	Array of strings	Specifies the list of objects to be notified if the alarm status changes. **NOTE** The IDs in the list are strings.

Table 5-22 ok_actions data structure description

Parameter	Type	Description
type	String	Specifies the notification type when an alarm is triggered. • **notification**: indicates that a notification will be sent. • **autoscaling**: indicates that a scaling action will be triggered.
notificationList	Array of strings	Specifies the ID list of objects to be notified if the alarm status changes. **NOTE** The IDs in the list are strings.

Table 5-23 condition data structure description

Parameter	Type	Description
period	Integer	Specifies the interval (seconds) for checking whether the configured alarm rules are met.
filter	String	Specifies the data rollup method. The following methods are supported: • **average**: Cloud Eye calculates the average value of metric data within a rollup period. • **max**: Cloud Eye calculates the maximum value of metric data within a rollup period. • **min**: Cloud Eye calculates the minimum value of metric data within a rollup period. • **sum**: Cloud Eye calculates the sum of metric data within a rollup period. • **variance**: Cloud Eye calculates the variance value of metric data within a rollup period.
comparison_operator	String	Specifies the alarm threshold operator, which can be >, =, <, ≥, or ≤.

Parameter	Type	Description
value	Double	Specifies the alarm threshold. Supported range: **0** to **Number. MAX_VALUE (1.7976931348623157e+108)** For detailed thresholds, see the value range of each metric in the appendix. For example, you can set ECS **cpu_util** in **Services Interconnected with Cloud Eye** to **80**.
unit	String	Specifies the data unit. Enter up to 32 characters.
count	Integer	Specifies the number of consecutive occurrence times that the alarm policy was met. Supported range: **1** to **5**

Table 5-24 meta_data data structure description

Parameter	Type	Description
count	Integer	Specifies the number of returned results.
marker	String	Specifies the pagination marker. For example, you have queried 10 records this time and **alarm_id** of the tenth record is **1441967036681YkazZ0deN**. In your next query, if **start** is set to **al1441967036681YkazZ0deN**, you can start your query from the next alarm rule ID of **al1441967036681YkazZ0deN**.
total	Integer	Specifies the total number of query results.

- Example response

```
{
    "metric_alarms": [
        {
            "alarm_name": "alarm-ttttttt",
            "alarm_description": "",
            "metric": {
                "namespace": "SYS.ECS",
                "dimensions": [
                    {
                        "name": "instance_id",
                        "value": "07814c0e-59a1-4fcd-a6fb-56f2f6923046"
                    }
                ],
                "metric_name": "cpu_util"
            },
            "condition": {
                "period": 300,
                "filter": "average",
                "comparison_operator": ">=",
                "value": 0,
                "unit": "%",
                "count": 3
```

```
        },
        "alarm_enabled": true,
        "alarm_level": 2,
        "alarm_action_enabled": false,
        "alarm_id": "al15330507498596W7vmlGKL",
        "update_time": 1533050749992,
        "alarm_state": "alarm"
    },
    {
        "alarm_name": "alarm-m5rwxxxxxxx",
        "alarm_description": "",
        "metric": {
            "namespace": "SYS.ECS",
            "dimensions": [
                {
                    "name": "instance_id",
                    "value": "30f3858d-4377-4514-9081-be5bdbf1392e"
                }
            ],
            "metric_name": "network_incoming_bytes_aggregate_rate"
        },
        "condition": {
            "period": 300,
            "filter": "average",
            "comparison_operator": ">=",
            "value": 12,
            "unit": "Byte/s",
            "count": 3
        },
        "alarm_enabled": true,
        "alarm_level": 2,
        "alarm_action_enabled": true,
        "alarm_actions": [
            {
                "type": "notification",
                "notificationList": [
                    "urn:smn:region:68438a86d98e427e907e0097b7e35d48:test0315"
                ]
            }
        ],
        "ok_actions": [
            {
                "type": "notification",
                "notificationList": [
                    "urn:smn:region:68438a86d98e427e907e0097b7e35d48:test0315"
                ]
            }
        ],
        "alarm_id": "al1533031226533nKJexAlbq",
        "update_time": 1533204036276,
        "alarm_state": "ok"
    }
  ],
  "meta_data": {
    "count": 2,
    "marker": "al1533031226533nKJexAlbq",
    "total": 389
  }
}
```


Returned Values

- Normal

  200
- Abnormal

Returned Value	Description
400 Bad Request	Request error.
401 Unauthorized	The authentication information is not provided or is incorrect.
403 Forbidden	You are forbidden to access the page requested.
408 Request Timeout	The request timed out.
429 Too Many Requests	Concurrent requests are excessive.
500 Internal Server Error	Failed to complete the request because of an internal service error.
503 Service Unavailable	The service is currently unavailable.

Error Codes

See **Error Codes**.

5.3.2 Querying an Alarm Rule

Function

This API is used to query an alarm rule based on the alarm rule ID.

URI

GET /V1.0/{project_id}/alarms/{alarm_id}

- Parameter description

  Table 5-25 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.
alarm_id	Yes	Specifies the alarm rule ID.

- Example

```
GET https://{Cloud Eye endpoint}/V1.0/{project_id}/alarms/al1441967036681YkazZ0deN
```

Request

None

Response

- Response parameters

Parameter	Type	Description
metric_alarms	Array of objects	Specifies the list of alarm objects. For details, see **Table 5-26**.

Table 5-26 metric_alarms data structure description

Parameter	Type	Description
alarm_name	String	Specifies the alarm rule name.
alarm_description	String	Provides supplementary information about the alarm rule.
metric	Object	Specifies the alarm metric. For details, see **Table 5-27**.
condition	Object	Specifies the alarm triggering condition. For details, see **Table 5-31**.
alarm_enabled	Boolean	Specifies whether to enable the alarm rule.
alarm_level	Integer	Specifies the alarm severity, which can be **1**, **2** (default), **3** or **4**, indicating critical, major, minor, and informational, respectively.
alarm_action_enabled	Boolean	Specifies whether to enable the action to be triggered by an alarm.
alarm_actions	Array of objects	Specifies the action to be triggered by an alarm. For details, see **Table 5-29**.
ok_actions	Array of objects	Specifies the action to be triggered after the alarm is cleared. For details, see **Table 5-30**.
alarm_id	String	Specifies the alarm rule ID.
update_time	Long	Specifies when the alarm status changed. The time is a UNIX timestamp and the unit is ms.
alarm_state	String	Specifies the alarm status, which can be • **ok**: The alarm status is normal. • **alarm**: An alarm is generated. • **insufficient_data**: The required data is insufficient.

Table 5-27 metric data structure description

Parameter	Type	Description
namespace	String	Query the namespace of a service. For details, see **Services Interconnected with Cloud Eye**.
dimensions	Array of objects	Specifies the list of metric dimensions. For details, see **Table 5-28**.
metric_name	String	Specifies the metric ID. For example, if the monitoring metric of an ECS is CPU usage, **metric_name** is **cpu_util**. For details, see **Services Interconnected with Cloud Eye**.

Table 5-28 dimensions data structure description

Parameter	Type	Description
name	String	Specifies the dimension. For example, the ECS dimension is **instance_id**. For details about the dimension of each service, see the **key** column in **Services Interconnected with Cloud Eye**.
value	String	Specifies the dimension value, for example, an ECS ID. Enter 1 to 256 characters.

Table 5-29 alarm_actions data structure description

Parameter	Type	Description
type	String	Specifies the alarm notification type. • **notification**: indicates that a notification will be sent. • **autoscaling**: indicates that a scaling action will be triggered.
notificationList	Array of strings	Specifies the list of objects to be notified if the alarm status changes. **NOTE** The IDs in the list are strings.

Table 5-30 ok_actions data structure description

Parameter	Type	Description
type	String	Specifies the notification type when an alarm is triggered. • **notification**: indicates that a notification will be sent. • **autoscaling**: indicates that a scaling action will be triggered.
notificationList	Array of strings	Specifies the list of objects to be notified if the alarm status changes. **NOTE** The IDs in the list are strings.

Table 5-31 condition data structure description

Parameter	Type	Description
period	Integer	Specifies the interval (seconds) for checking whether the configured alarm rules are met.
filter	String	Specifies the data rollup method. The following methods are supported: • **average**: Cloud Eye calculates the average value of metric data within a rollup period. • **max**: Cloud Eye calculates the maximum value of metric data within a rollup period. • **min**: Cloud Eye calculates the minimum value of metric data within a rollup period. • **sum**: Cloud Eye calculates the sum of metric data within a rollup period. • **variance**: Cloud Eye calculates the variance value of metric data within a rollup period.
comparison_operator	String	Specifies the alarm threshold operator, which can be >, =, <, ≥, or ≤.
value	Double	Specifies the alarm threshold. Supported range: **0** to **Number. MAX_VALUE (1.7976931348623157e+108)** For detailed thresholds, see the value range of each metric in the appendix. For example, you can set ECS **cpu_util** in **Services Interconnected with Cloud Eye** to **80**.
unit	String	Specifies the data unit. Enter up to 32 characters.

Parameter	Type	Description
count	Integer	Specifies the number of consecutive occurrence times that the alarm policy was met. Supported range: **1** to **5**

- Example response

```
{
"metric_alarms":
 [
  {
   "alarm_name":"alarm-ipwx",
   "alarm_description":"",
   "metric":
    {
     "namespace":"SYS.ELB",
     "dimensions":
     [
      {
       "name":"lb_instance_id",
       "value":"44d06d10-bce0-4237-86b9-7b4d1e7d5621"
      }
     ],
     "metric_name":"m8_out_Bps"
     },
   "condition":
    {
     "period":300,
     "filter":"sum",
     "comparison_operator":">=",
     "value":0,
     "unit":"",
     "count":1
     },
   "alarm_enabled":true,
   "alarm_level": 2,
   "alarm_action_enabled":true,
   "alarm_actions":
    [
     {
      "type":"notification",
      "notificationList":["urn:smn:region:68438a86d98e427e907e0097b7e35d48:sd"]
     }
    ],
   "ok_actions":
    [
     {
      "type":"notification",
      "notificationList":["urn:smn:region:68438a86d98e427e907e0097b7e35d48:sd"]
     }
    ],
   "alarm_id":"al1498096535573r8DNy7Gyk",
   "update_time":1498100100000,
   "alarm_state":"alarm"
  }
 ]
}
```

Returned Values

- Normal

  200

- Abnormal

Returned Value	Description
400 Bad Request	Request error.
401 Unauthorized	The authentication information is not provided or is incorrect.
403 Forbidden	You are forbidden to access the page requested.
408 Request Timeout	The request timed out.
429 Too Many Requests	Concurrent requests are excessive.
500 Internal Server Error	Failed to complete the request because of an internal service error.
503 Service Unavailable	The service is currently unavailable.

Error Codes

See **Error Codes**.

5.3.3 Enabling or Disabling an Alarm Rule

Function

This API is used to enable or disable an alarm rule.

URI

PUT /V1.0/{project_id}/alarms/{alarm_id}/action

- Parameter description

  Table 5-32 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.
alarm_id	Yes	Specifies the alarm rule ID.

- Example

```
PUT https://{Cloud Eye endpoint}/V1.0/{project_id}/alarms/al1441967036681YkazZ0deN/action
```

Request

- Request parameters

Table 5-33 Request parameters

Parameter	Mandatory	Type	Description
alarm_enabled	Yes	Boolean	Specifies whether the alarm rule is enabled. • **true**: indicates that the alarm rule is enabled. • **false**: indicates that the alarm rule is disabled.

- Example request

```
{
   "alarm_enabled":true
}
```

Response

The response has no message body.

Returned Values

- Normal

  204

- Abnormal

Returned Value	Description
400 Bad Request	Request error.
401 Unauthorized	The authentication information is not provided or is incorrect.
403 Forbidden	Access to the requested page is forbidden.
408 Request Timeout	The request timed out.
429 Too Many Requests	Concurrent requests are excessive.
500 Internal Server Error	Failed to complete the request because of an internal service error.
503 Service Unavailable	The service is currently unavailable.

Error Codes

See **Error Codes**.

5.3.4 Deleting an Alarm Rule

Function

This API is used to delete an alarm rule.

URI

DELETE /V1.0/{project_id}/alarms/{alarm_id}

- Parameter description

  Table 5-34 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.
alarm_id	Yes	Specifies the alarm rule ID.

- Example
  DELETE https://{Cloud Eye endpoint}/V1.0/{project_id}/alarms/al1441967036681YkazZ0deN

Request

The request has no message body.

Response

The response has no message body.

Returned Values

- Normal

  204

- Abnormal

Returned Value	Description
400 Bad Request	Request error.
401 Unauthorized	The authentication information is not provided or is incorrect.
403 Forbidden	Access to the requested page is forbidden.
408 Request Timeout	The request timed out.
429 Too Many Requests	Concurrent requests are excessive.

Returned Value	Description
500 Internal Server Error	Failed to complete the request because of an internal service error.
503 Service Unavailable	The service is currently unavailable.

Error Codes

See **Error Codes**.

5.3.5 Creating an Alarm Rule

Function

This API is used to create an alarm rule.

URI

POST /V1.0/{project_id}/alarms

- Parameter description

  Table 5-35 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.

- Example

```
POST https://{Cloud Eye endpoint}/V1.0/{project_id}/alarms
```

Request

- Request parameters

  Table 5-36 Request parameters

Parameter	Mandatory	Type	Description
alarm_name	Yes	String	Specifies the alarm rule name. Enter 1 to 128 characters. Only letters, digits, underscores (_), and hyphens (-) are allowed.

Parameter	Mandatory	Type	Description
alarm_description	No	String	Provides supplementary information about the alarm rule. Enter 0 to 256 characters.
metric	Yes	Object	Specifies the alarm metric. For details, see **Table 5-37**.
condition	Yes	Object	Specifies the alarm triggering condition. For details, see **Table 5-41**.
alarm_enabled	No	Boolean	Specifies whether to enable the alarm. The default value is **true**.
alarm_action_enabled	No	Boolean	Specifies whether to enable the action to be triggered by an alarm. The default value is **true**. **NOTE** If you set **alarm_action_enabled** to **true**, you must specify either **alarm_actions** or **ok_actions**. (You do not need to configure the deprecated parameter **insufficientdata_actions**.) If **alarm_actions** and **ok_actions** coexist, their **notificationList** must be the same. (You do not need to configure the deprecated parameter **insufficientdata_actions**.)
alarm_level	No	Integer	Specifies the alarm severity, which can be **1**, **2** (default), **3** or **4**, indicating critical, major, minor, and informational, respectively.
alarm_type	No	String	Specifies the alarm rule type. **EVENT.SYS**: The alarm rule is created for system events. **EVENT.CUSTOM**: The alarm rule is created for custom events.

Parameter	Mandatory	Type	Description

alarm_actions	No	Arrays of objects	Specifies the action to be triggered by an alarm.
			An example structure is as follows:
			{
			"type": "notification","notificationList" : ["urn:smn:region: 68438a86d98e427e907e0097b 7e35d47:sd"]
			}
			For details, see **Table 5-39**.

ok_actions	No	Arrays of objects	Specifies the action to be triggered after the alarm is cleared. Its structure is: { "type": "notification","notificationList" : ["urn:smn:region: 68438a86d98e427e907e0097b 7e35d47:sd"] } For details, see **Table 5-40**.

Table 5-37 metric data structure description

Parameter	Mandatory	Type	Description
namespace	Yes	String	Specifies the namespace of a service. For details, see **Services Interconnected with Cloud Eye**. The namespace must be in the **service.item** format and contain 3 to 32 characters. **service** and **item** each must start with a letter and contain only letters, digits, and underscores (_).
dimensions	No	Arrays of objects	Specifies the metric dimension list. When **resource_group_id** is not used, **dimensions** is mandatory. For details, see **Table 5-38**.

Parameter	Mandatory	Type	Description
metric_name	Yes	String	Specifies the metric name. Start with a letter. Enter 1 to 64 characters. Only letters, digits, and underscores (_) are allowed. For details, see the metric name queried in **Querying Metrics**.
resource_group_id	No	String	Specifies the resource group ID selected during the alarm rule creation, for example, **rg1603786526428bWbVmk4rP**. **NOTE** If you create alarm rules for resource groups, you must specify **resource_group_id** and **name**, enter at least one dimension for **dimensions**, and set **alarm_type** to **RESOURCE_GROUP**.

Table 5-38 dimensions data structure description

Parameter	Mandatory	Type	Description
name	Yes	String	Specifies the dimension. For example, the ECS dimension is **instance_id**. For details about the dimension of each service, see the **key** column in **Services Interconnected with Cloud Eye**. Start with a letter. Enter 1 to 32 characters. Only letters, digits, underscores (_), and hyphens (-) are allowed.
value	Yes	String	Specifies the dimension value, for example, an ECS ID. Start with a letter or a digit. Enter 1 to 256 characters. Only letters, digits, underscores (_), and hyphens (-) are allowed.

Table 5-39 alarm_actions data structure description

Parameter	Mandatory	Type	Description
type	Yes	String	Specifies the alarm notification type. • **notification**: indicates that a notification will be sent. • **autoscaling**: indicates that a scaling action will be triggered.
notificationList	Yes	Arrays of strings	Specifies the list of objects to be notified if the alarm status changes. You can configure up to 5 object IDs. **topicUrn** can be obtained from SMN. For details, see **Querying Topics**. If you set **type** to **notification**, you must specify **notificationList**. If you set **type** to **autoscaling**, you must set **notificationList** to **[]**. **NOTE** • To make the Auto Scaling (AS) alarm rule take effect, you must bind the scaling policy. For details, see **Creating an AS Policy**. • If you set **alarm_action_enabled** to **true**, you must specify either **alarm_actions** or **ok_actions**. (You do not need to configure the deprecated parameter **insufficientdata_actions**.) • If **alarm_actions** and **ok_actions** coexist, their **notificationList** must be the same. (You do not need to configure the deprecated parameter **insufficientdata_actions**.) • The IDs in the list are strings.

Table 5-40 ok_actions data structure description

Parameter	Mandatory	Type	Description
type	Yes	String	Specifies the notification type when an alarm is triggered. • **notification**: indicates that a notification will be sent. • **autoscaling**: indicates that a scaling action will be triggered.

Parameter	Mandatory	Type	Description
notificationList	Yes	Arrays of objects	Specifies the list of objects to be notified if the alarm status changes. The list contains a maximum of 5 object IDs. **topicUrn** can be obtained from SMN. For details, see **Querying Topics**. **NOTE** If you set **alarm_action_enabled** to **true**, you must specify either **alarm_actions** or **ok_actions**. (You do not need to configure the deprecated parameter **insufficientdata_actions**.) If **alarm_actions** and **ok_actions** coexist, their **notificationList** must be the same. (You do not need to configure the deprecated parameter **insufficientdata_actions**.)

Table 5-41 condition data structure description

Parameter	Mandatory	Type	Description
period	Yes	Integer	Specifies the period during which Cloud Eye determines whether to trigger an alarm. Unit: second Possible periods are **1**, **300**, **1200**, **3600**, **14400**, and **86400**. **NOTE** • If you set **period** to **1**, Cloud Eye uses raw data to determine whether to trigger an alarm.
filter	Yes	String	Specifies the data rollup method. Possible methods are **max**, **min**, **average**, **sum**, or **variance**.
comparison_operator	Yes	String	Specifies the operator of alarm thresholds. Possible operators are >, =, <, >=, and <=.
value	Yes	Double	Specifies the alarm threshold. Supported range: **0** to **Number. MAX_VALUE (1.7976931348623157e+108)** For detailed thresholds, see the value range of each metric in the appendix. For example, you can set ECS **cpu_util** in **Services Interconnected with Cloud Eye** to **80**.
unit	No	String	Specifies the data unit. Enter up to 32 characters.

Parameter	Mandatory	Type	Description
count	Yes	Integer	Specifies the number of consecutive occurrence times that the alarm policy was met. Supported range: **1** to **5**

- Example request

```
{
   "alarm_name": "alarm-rp0E",
   "alarm_description": "",
   "metric": {
      "namespace": "SYS.ECS",
      "dimensions": [
         {
            "name": "instance_id",
            "value": "33328f02-3814-422e-b688-bfdba93d4051"
         }
      ],
      "metric_name": "network_outgoing_bytes_rate_inband"
   },
   "condition": {
      "period": 300,
      "filter": "average",
      "comparison_operator": ">=",
      "value": 6,
      "unit": "Byte/s",
      "count": 1
  },
   "alarm_enabled": true,
   "alarm_action_enabled": true,
   "alarm_level": 2,
   "alarm_actions": [
      {
         "type": "notification",
         "notificationList": ["urn:smn:region:68438a86d98e427e907e0097b7e35d48:sd"]
      }
   ],
   "ok_actions": [
      {
         "type": "notification",
         "notificationList": ["urn:smn:region:68438a86d98e427e907e0097b7e35d48:sd"]
      }
   ]
}
```

Response

- Response parameters

Table 5-42 Response parameters

Parameter	Type	Description
alarm_id	String	Specifies the alarm rule ID.

- Example response

```
{
   "alarm_id":"al1450321795427dR8p5mQBo"
}
```


Returned Values

- Normal

  201
- Abnormal

Returned Value	Description
400 Bad Request	Request error.
401 Unauthorized	The authentication information is not provided or is incorrect.
403 Forbidden	Access to the requested page is forbidden.
408 Request Timeout	The request timed out.
429 Too Many Requests	Concurrent requests are excessive.
500 Internal Server Error	Failed to complete the request because of an internal service error.
503 Service Unavailable	The service is currently unavailable.

Error Codes

See **Error Codes**.

5.4 Monitoring Data Management

5.4.1 Querying Monitoring Data

Function

This API is used to query the monitoring data at a specified granularity for a specified metric in a specified period of time. You can specify the dimension of data to be queried.

URI

GET /V1.0/{project_id}/metric-data?
namespace={namespace}&metric_name={metric_name}&dim.
{i}=key,value&from={from}&to={to}&period={period}&filter={filter}

- Parameter description

Table 5-43 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.

Table 5-44 Query parameter description

Parameter	Mandatory	Type	Description
namespace	Yes	String	Specifies the namespace of a service. For details, see **Services Interconnected with Cloud Eye**. The namespace must be in the **service.item** format and contain 3 to 32 characters. **service** and **item** each must start with a letter and contain only letters, digits, and underscores (_).
metric_name	Yes	String	Specifies the metric name. You can obtain the metric names of existing alarm rules by referring to **Querying Metrics**.
from	Yes	String	Specifies the start time of the query. The time is a UNIX timestamp and the unit is ms. Rollup aggregates the raw data generated within a period to the start time of the period. Therefore, if **from** and **to** are within a period, the query result will be empty due to the rollup failure. Set **from** to at least one period earlier than the current time. Take the 5-minute period as an example. If it is 10:35 now, the raw data generated between 10:30 and 10:35 will be aggregated to 10:30. Therefore, in this example, if **period** is 5 minutes, **from** should be 10:30. **NOTE** Cloud Eye rounds up **from** based on the level of granularity required to perform the rollup.

Parameter	Mandatory	Type	Description
to	Yes	String	Specifies the end time of the query. The time is a UNIX timestamp and the unit is ms. **from** must be earlier than **to**.
period	Yes	Integer	Specifies how often Cloud Eye aggregates data, which can be • **1**: Cloud Eye performs no aggregation and displays raw data. • **300**: Cloud Eye aggregates data every 5 minutes. • **1200**: Cloud Eye aggregates data every 20 minutes. • **3600**: Cloud Eye aggregates data every 1 hour. • **14400**: Cloud Eye aggregates data every 4 hours. • **86400**: Cloud Eye aggregates data every 24 hours.
filter	Yes	String	Specifies the data rollup method, which can be • **average**: Cloud Eye calculates the average value of metric data within a rollup period. • **max**: Cloud Eye calculates the maximum value of metric data within a rollup period. • **min**: Cloud Eye calculates the minimum value of metric data within a rollup period. • **sum**: Cloud Eye calculates the sum of metric data within a rollup period. • **variance**: Cloud Eye calculates the variance value of metric data within a rollup period. **NOTE** Rollup uses a rollup method to aggregate raw data generated within a specific period. Take the 5-minute period as an example. If it is 10:35 now, the raw data generated between 10:30 and 10:35 will be aggregated to 10:30.

Parameter	Mandatory	Type	Description
dim	Yes	String	A maximum of three metric dimensions are supported, and the dimensions are numbered from 0 in the **dim.{i}=key,value** format. **key** cannot exceed 32 characters and **value** cannot exceed 256 characters. The following dimensions are only examples. For details about whether multiple dimensions are supported, see the dimension description in the **monitoring indicator description of each service**. Single dimension: **dim.0=instance_id,i-12345** Multiple dimensions: **dim.0=instance_id,i-12345&dim.1=instance_name,i-1234**

NOTE

- **dimensions** can be obtained from the response body by calling the API for **Querying Metrics**.
- OBS metric data can be queried only when the related OBS APIs are called.

- Example:

  Request example 1: View the CPU usage of ECS whose ID is **6f3c6f91-4b24-4e1b-b7d1-a94ac1cb011d** from 2019-04-30 20:00:00 to 2019-04-30 22:00:00. The monitoring interval is 20 minutes.

  ```
  GET https://{Cloud Eye endpoint}/V1.0/{project_id}/metric-data?namespace=SYS.ECS&metric_name=cpu_util&dim.0=instance_id,6f3c6f91-4b24-4e1b-b7d1-a94ac1cb011d&from=1556625600000&to=1556632800000&period=1200&filter=min
  ```

Request

None

Response

- Response parameters

Table 5-45 Response parameters

Parameter	Type	Description
datapoints	Array of objects	Specifies the metric data list. For details, see **Table 5-46**. Since Cloud Eye rounds up **from** based on the level of granularity for data query, **datapoints** may contain more data points than expected.
metric_name	String	Specifies the metric ID. For example, if the monitoring metric of an ECS is CPU usage, **metric_name** is **cpu_util**. For details, see **Services Interconnected with Cloud Eye**.

Table 5-46 datapoints data structure description

Parameter	Type	Description
average	Double	Specifies the average value of metric data within a rollup period.
max	Double	Specifies the maximum value of metric data within a rollup period.
min	Double	Specifies the minimum value of metric data within a rollup period.
sum	Double	Specifies the sum of metric data within a rollup period.
variance	Double	Specifies the variance of metric data within a rollup period.
timestamp	Long	Specifies when the metric is collected. It is a UNIX timestamp in milliseconds.
unit	String	Specifies the metric unit.

- Example response

  Example response 1: The dimension is SYS.ECS, and the average CPU usage of ECSs is displayed.

```
{
    "datapoints": [
        {
            "average": 0.23,
            "timestamp": 1442341200000,
            "unit": "%"
        }
    ],
    "metric_name": "cpu_util"
}
```

  Example response 2: The dimension is SYS.ECS, and the sum CPU usage of ECSs is displayed.

```
{
    "datapoints": [
```



```
        {
            "sum": 0.53,
            "timestamp": 1442341200000,
            "unit": "%"
        }
    ],
    "metric_name": "cpu_util"
}
```

Example response 3: The dimension is SYS.ECS, and the maximum CPU usage of ECSs is displayed.

```
{
    "datapoints": [
        {
            "max": 0.13,
            "timestamp": 1442341200000,
            "unit": "%"
        }
    ],
    "metric_name": "cpu_util"
}
```

Returned Values

- Normal

  200

- Abnormal

Returned Value	Description
400 Bad Request	Request error.
401 Unauthorized	The authentication information is not provided or is incorrect.
403 Forbidden	You are forbidden to access the page requested.
408 Request Timeout	The request timed out.
429 Too Many Requests	Concurrent requests are excessive.
500 Internal Server Error	Failed to complete the request because of an internal service error.
503 Service Unavailable	The service is currently unavailable.

Error Codes

See **Error Codes**.

5.4.2 Adding Monitoring Data

Function

This API is used to add one or more pieces of custom metric monitoring data to solve the problem that the system metrics cannot meet specific service requirements.

URI

POST /V1.0/{project_id}/metric-data

- Parameter description

Table 5-47 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.

Request

NOTICE

1. The size of a POST request cannot exceed 512 KB. Otherwise, the request will be denied.
2. The period for sending POST requests must be shorter than the minimum aggregation period. Otherwise, the aggregated data will be noncontinuous. For example, if the aggregation period is 5 minutes and the POST request sending period is 7 minutes, the data will be aggregated every 10 minutes, rather than 5 minutes.
3. Timestamp (collect_time) in the POST request body value must be within the period that starts from three days before the current time to 10 minutes after the current time. If it is not in this range, you are not allowed to insert the metric data.

- Request parameters

Table 5-48 Parameter description

Parameter	Type	Mandatory	Description
Array elements	Array of objects	Yes	Specifies whether to add one or more pieces of custom metric monitoring data. For details, see **Table 5-49**.

Table 5-49 Array elements

Parameter	Mandatory	Type	Description
metric	Yes	Object	Specifies the metric data. For details, see **Table 5-50**.
ttl	Yes	Integer	Specifies the data validity period. The unit is second. Supported range: 1 to 604800 If the validity period expires, the data will be automatically deleted.
collect_time	Yes	Long	Specifies when the data was collected. The time is UNIX timestamp (ms) format. **NOTE** Since there is a latency between the client and the server, the data timestamp to be inserted should be within the period that starts from three days before the current time plus 20s to 10 minutes after the current time minus 20s. In this way, the timestamp will be inserted to the database without being affected by the latency.
value	Yes	Double	Specifies the monitoring metric data to be added, which can be an integer or a floating point number.
unit	No	String	Specifies the data unit. Enter a maximum of 32 characters.
type	No	String	Specifies the enumerated type. Possible types: • **int** • **float**

Table 5-50 metric data structure description

Parameter	Mandatory	Type	Description
namespace	Yes	String	Specifies the customized namespace. For details, see **Services Interconnected with Cloud Eye**. The namespace must be in the **service.item** format and contain 3 to 32 characters. **service** and **item** each must start with a letter and contain only letters, digits, and underscores (_). In addition, **service** cannot start with **SYS**, **AGT**, or **SRE**, and **namespace** cannot be **SERVICE.BMS** because this namespace has been used by the system. You can leave this parameter blank when you set **alarm_type** to **(EVENT.SYS| EVENT.CUSTOM)**.
dimensions	Yes	Array of objects	Specifies the metric dimension. A maximum of three dimensions are supported. For details, see **Table 5-51**.
metric_name	Yes	String	Specifies the metric ID. For example, if the monitoring metric of an ECS is CPU usage, **metric_name** is **cpu_util**. For details, see **Services Interconnected with Cloud Eye**.

Table 5-51 dimensions data structure description

Parameter	Mandatory	Type	Description
name	Yes	String	Specifies the dimension. For example, the ECS dimension is **instance_id**. For details about the dimension of each service, see the **key** column in **Services Interconnected with Cloud Eye**. Start with a letter. Enter 1 to 32 characters. Only letters, digits, underscores (_), and hyphens (-) are allowed.

Parameter	Mandatory	Type	Description
value	Yes	String	Specifies the dimension value, for example, an ECS ID. Start with a letter or a digit. Enter 1 to 256 characters. Only letters, digits, underscores (_), and hyphens (-) are allowed.

- Example request

  Example request 1: Add **cpu_util** data of a custom dimension. The instance ID is **6f3c6f91-4b24-4e1b-b7d1-a94ac1cb011d**.

```
[
  {
    "metric": {
      "namespace": "MINE.APP",
      "dimensions": [
        {
          "name": "instance_id",
          "value": "6f3c6f91-4b24-4e1b-b7d1-a94ac1cb011d"
        }
      ],
      "metric_name": "cpu_util"
    },
    "ttl": 172800,
    "collect_time": 1463598260000,
    "type": "float",
    "value": 0.09,
    "unit": "%"
  },
  {
    "metric": {
      "namespace": "MINE.APP",
      "dimensions": [
        {
          "name": "instance_id",
          "value": "6f3c6f91-4b24-4e1b-b7d1-a94ac1cb011d"
        }
      ],
      "metric_name": "cpu_util"
    },
    "ttl": 172800,
    "collect_time": 1463598270000,
    "type": "float",
    "value": 0.12,
    "unit": "%"
  }
]
```

  Example request 2: Add **rds021_myisam_buf_usage** data of the RDS instance whose **rds_cluster_id** is **3c8cc15614ab46f5b8743317555e0de2in01**.

```
[
  {
    "metric": {
      "namespace": "SYS.RDS",
      "dimensions": [
        {
          "name": "rds_cluster_id",
          "value": "3c8cc15614ab46f5b8743317555e0de2in01"
        }
      ],
      "metric_name": "rds021_myisam_buf_usage"
    },
```



```
        "ttl": 172800,
        "collect_time": 1463598260000,
        "type": "float",
        "value": 0.01,
        "unit": "Ratio"
    }
]
```

Response

The response has no message body.

Returned Values

- Normal

  201

- Abnormal

Returned Value	Description
400 Bad Request	Request error.
401 Unauthorized	The authentication information is not provided or is incorrect.
403 Forbidden	You are forbidden to access the page requested.
408 Request Timeout	The request timed out.
429 Too Many Requests	Concurrent requests are excessive.
500 Internal Server Error	Failed to complete the request because of an internal service error.
503 Service Unavailable	The service is currently unavailable.

Error Codes

See **Error Codes**.

5.5 Quota Management

5.5.1 Querying Quotas

Function

This API is used to query a resource quota and the used amount. The current resource refers to alarm rules only.

URI

GET /V1.0/{project_id}/quotas

- Parameter description

  Table 5-52 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.

- Example: Query the alarm rule quota.

```
GET https://{Cloud Eye endpoint}/V1.0/{project_id}/quotas
```

Request

None

Response

- Response parameters

  Table 5-53 Response parameters

Parameter	Type	Description
quotas	Object	Specifies the quota list. For details, see **Table 5-54**.

  Table 5-54 Data structure description of **quotas**

Parameter	Type	Description
resources	Array of objects	Specifies the resource quota list. For details, see **Table 5-55**.

Table 5-55 Data structure description of **resources**

Parameter	Type	Description
type	String	Specifies the quota type. **alarm** indicates the alarm rule.
used	Integer	Specifies the used amount of the quota.
unit	String	Specifies the quota unit.
quota	Integer	Specifies the total amount of the quota.

- Example response

```
{
"quotas":
  {
  "resources": [
        {
            "unit":"",
            "type":"alarm",
            "quota":1000,
            "used":10
        }
    ]
  }
}
```

Returned Values

- Normal

  200

- Abnormal

Returned Value	Description
400 Bad Request	Request error.
401 Unauthorized	The authentication information is not provided or is incorrect.
403 Forbidden	Access to the requested page is forbidden.
408 Request Timeout	The request timed out.
429 Too Many Requests	Concurrent requests are excessive.
500 Internal Server Error	Failed to complete the request because of an internal service error.
503 Service Unavailable	The service is currently unavailable.

Error Codes

See **Error Codes**.

5.6 Event Monitoring

5.6.1 Reporting Events

Function

An API for reporting custom events is provided, which helps you collect and report abnormal events or important change events to Cloud Eye.

URI

POST /V1.0/{project_id}/events

- Parameter description

  Table 5-56 Parameter description

Parameter	Mandatory	Description
project_id	Yes	Specifies the project ID. For details about how to obtain the project ID, see **Obtaining a Project ID**.

- Example
  POST https://{Cloud Eye endpoint}/V1.0/{project_id}/events

Request

- Request parameters

  Table 5-57 Parameter description

Parameter	Type	Mandatory	Description
event_item	Arrays of objects	Yes	Specifies the event list. For details, see **Table 5-58**.

  Table 5-58 Parameter description of the **event_item** field

Parameter	Mandatory	Type	Description
event_name	Yes	String	Specifies the event name. Start with a letter. Enter 1 to 64 characters. Only letters, digits, and underscores (_) are allowed.

Parameter	Mandatory	Type	Description
event_source	Yes	String	Specifies the event source. The format is service.item. Set this parameter based on the site requirements. **service** and **item** each must be a string that starts with a letter and contains 3 to 32 characters, including only letters, digits, and underscores (_).
time	Yes	Long	Specifies when the event occurred, which is a UNIX timestamp (ms). **NOTE** Since there is a latency between the client and the server, the data timestamp to be inserted should be within the period that starts from one hour before the current time plus 20s to 10 minutes after the current time minus 20s. In this way, the timestamp will be inserted to the database without being affected by the latency. For example, if the current time is 2020.01.30 12:00:30, the timestamp inserted must be within the range [2020.01.30 11:00:50, 2020.01.30 12:10:10]. The corresponding UNIX timestamp is [1580353250, 1580357410].
detail	Yes	Arrays of objects	Specifies the event details. For details, see **Table 5-59**.

Table 5-59 detail data structure description

Parameter	Mandatory	Type	Description
content	No	String	Specifies the event content. Enter up to 4096 characters.
resource_id	No	String	Specifies the resource ID. Enter up to 128 characters, including letters, digits, underscores (_), hyphens (-), and colon (:). Example: 6a69bf28-ee62-49f3-9785-845dacd799ec To query the resource ID, perform the following steps: 1. Log in to the management console. 2. Under **Computing**, select **Elastic Cloud Server**. On the **Resource Overview** page, obtain the resource ID.

Parameter	Mandatory	Type	Description
resource_name	No	String	Specifies the resource name. Enter up to 128 characters, including letters, digits, underscores (_), and hyphens (-).
event_state	No	String	Specifies the event status. Valid value can be **normal**, **warning**, or **incident**.
event_level	No	String	Specifies the event severity. Its value can be **Critical**, **Major**, **Minor**, or **Info**.
event_user	No	String	Specifies the event user. Enter up to 64 characters, including letters, digits, underscores (_), hyphens (-), slashes (/), and spaces.

- Example request

```
[{
   "event_name":"systemInvaded",
   "event_source":"financial.System",
   "time":1522121194000,
   "detail":{
      "content":"The financial system was invaded",
      "group_id":"rg15221211517051YWWkEnVd",
      "resource_id":"1234567890sjgggad",
      "resource_name":"ecs001",
      "event_state":"normal",
      "event_level":"Major",
      "event_user":"xiaokong"
   }
},
{
   "event_name":"systemInvaded",
   "event_source":"financial.System",
   "time":1522121194020,
   "detail":{
      "content":"The financial system was invaded",
      "group_id":"rg15221211517051YWWkEnVd",
      "resource_id":"1234567890sjgggad",
      "resource_name":"ecs001",
      "event_state":"normal",
      "event_level":"Major",
      "event_user":"xihong"
   }
}]
```

Response

- Response parameters

Table 5-60 Parameter description

Parameter	Type	Description
Array elements	Arrays of objects	Specifies the event list. For details, see **Table 5-61**.

Table 5-61 Response parameters

Parameter	Mandatory	Type	Description
event_id	Yes	String	Specifies the event ID.
event_name	Yes	String	Specifies the event name. Start with a letter. Enter 1 to 64 characters. Only letters, digits, and underscores (_) are allowed.

- Example response

```
[
    {
        "event_id":"evdgiqwgedkkcvhdjcdu346",
        "event_name":"systemInvaded"
     },
     {
        "event_id":"evdgiqwgedkkcvhdjcdu347",
        "event_name":"systemParalysis"
     }
]
```

Returned Values

- Normal

  201

- Abnormal

Returned Value	Description
400 Bad Request	Request error.
401 Unauthorized	The authentication information is not provided or is incorrect.
403 Forbidden	Access to the requested page is forbidden.
408 Request Timeout	The request timed out.
429 Too Many Requests	Concurrent requests are excessive.
500 Internal Server Error	Failed to complete the request because of an internal service error.

Returned Value	Description
503 Service Unavailable	The service is currently unavailable.

Error Codes

See **Error Codes**.

6 Permissions Policies and Supported Actions

6.1 Supported Actions of the Metric Management API

Permission	API	Action	IAM Project	Enterprise Project
Query the metric list. You can specify the namespace, metric name, dimension, sorting order, start records, and the maximum number of records when using this API to query metrics.	GET /V1.0/{project_id}/metrics	ces:metrics:list	√	×

6.2 Supported Actions of the Alarm Rule Management APIs

Permission	API	Action	IAM Project	Enterprise Project
Query the alarm rule list. You can specify the paging parameters to limit the number of query results displayed on a page. You can also set the sorting order of query results.	GET /V1.0/{project_id}/alarms	ces:alarms:list	√	√
Query an alarm rule based on the alarm rule ID.	GET /V1.0/{project_id}/alarms/{alarm_id}	ces:alarms:get	√	√
Enable or disable an alarm rule.	PUT /V1.0/{project_id}/alarms/{alarm_id}/action	ces:alarmsOnOff:put	√	√
Delete an alarm rule.	DELETE /V1.0/{project_id}/alarms/{alarm_id}	ces:alarms:delete	√	√
Create an alarm rule.	POST /V1.0/{project_id}/alarms	ces:alarms:create	√	√

6.3 Supported Actions of the Monitoring Data Management APIs

Permission	API	Action	IAM Project	Enterprise Project
Query the monitoring data at a specified granularity for a specified metric in a specified period of time. You can specify the dimension of data to be queried.	GET /V1.0/{project_id}/metric-data?namespace={namespace}&metric_name={metric_name}&dim.{i}=key,value&from={from}&to={to}&period={period}&filter={filter}	ces:metricData:list	√	×
Add one or more pieces of custom metric monitoring data to solve the problem that the system metrics cannot meet specific service requirements.	POST /V1.0/{project_id}/metric-data	ces:metricData:create	√	×
Query the host configuration for a specified event type in a specified period of time. You can specify the dimension of data to be queried. (This API is provided for SAP Monitor to query the host configuration in the HANA scenario. In other scenarios, the host configuration cannot be queried with this API.)	GET /V1.0/{project_id}/event-data	ces:sapEventData:list	√	×

6.4 Supported Actions of the Quota Management API

Permission	API	Action	IAM Project	Enterprise Project
Query a resource quota and the used amount. Currently, the resource refers to alarm rules only.	GET /V1.0/{project_id}/quotas	ces:quotas:get	√	×

6.5 Supported Actions of the Event Monitoring API

Permission	API	Action	IAM Project	Enterprise Project
Report custom events.	POST /V1.0/{project_id}/events	ces:events:post	√	×

7 Common Parameters

7.1 Status Codes

- Normal

Returned Value	Description
200 OK	The results of GET and PUT operations are returned as expected.
201 Created	The results of the POST operation are returned as expected.
202 Accepted	The request has been accepted for processing.
204 No Content	The results of the DELETE operation are returned as expected.

- Abnormal

Returned Value	Description
400 Bad Request	The server failed to process the request.
401 Unauthorized	You must enter a username and password to access the requested page.
403 Forbidden	You are forbidden to access the requested page.
404 Not Found	The server cannot find the requested page.
405 Method Not Allowed	You are not allowed to use the method specified in the request.
406 Not Acceptable	The response generated by the server cannot be accepted by the client.

Returned Value	Description
407 Proxy Authentication Required	You must use the proxy server for authentication so that the request can be processed.
408 Request Timeout	The request timed out.
409 Conflict	The request could not be processed due to a conflict.
500 Internal Server Error	Failed to complete the request because of a service error.
501 Not Implemented	Failed to complete the request because the server does not support the requested function.
502 Bad Gateway	Failed to complete the request because the request is invalid.
503 Service Unavailable	Failed to complete the request. The service is unavailable.
504 Gateway Timeout	A gateway timeout error occurred.

7.2 Error Codes

Function

If an error occurs during API calling, the system returns error information. This section describes the error codes contained in the error information for Cloud Eye APIs.

Example Response

```
{
    "code": 400,
    "element": "Bad Request",
    "message": "The system received a request which cannot be recognized",
    "details": {
        "details": "Some content in message body is not correct",
        "code": "ces.0014"
    }
}
```

Glossary

Glossary	Description
Cloud Eye	Cloud Eye
Built-in metric	Each service has its own built-in metrics and dimensions. For example, an (SYS.ECS) supports **cpu_util**.

Glossary	Description
Metric	A metric consists of the namespace, dimension (optional), and metric name. A metric name solely does not identify any object.

Error Code Description

Module	HTTP Status Code	Error Code	Error Code Description	Error Message	Measure
Cloud Eye	500	ces.0007	Internal service error	Internal service error.	Contact technical support.
API	400	ces.0001	The request content cannot be empty.	The content must be specified.	Specify the request content.
	400	ces.0003	The project ID is left blank or is incorrect.	The tenant ID is left blank or incorrect.	Add or use the correct tenant ID.
	400	ces.0004	The API version is not specified.	The API version must be specified.	Specify the API version in the request URL.
	400	ces.0005	The API version is incorrect.	The API version is incorrect.	Use the correct API version.
	400	ces.0006	The paging address is incorrect.	The paging address is incorrect.	Use correct pagination information.
	403	ces.0009	System metrics cannot be added.	Adding SYS metric is not allowed	Use correct rights to add metrics.
	403	ces.0010	System metrics cannot be deleted.	Deleting SYS metric is not allowed	Use correct rights to delete metrics.
	400	ces.0011	The request is invalid.	The request is invalid.	Check the request.

Module	HTTP Status Code	Error Code	Error Code Description	Error Message	Measure
	400	ces.0013	The URL parameter is invalid or does not exist.	The URL parameter is invalid or does not exist.	Check the URL parameter.
	400	ces.0014	Some content in the message body is correct.	Some content in message body is not correct.	Check the request body parameters.
	401	ces.0015	Authentication fails or valid authentication information is not provided.	Authentication fails or the authentication information is not provided.	Check whether the user name or password (or AK or SK) for obtaining the token is correct.
	404	ces.0016	The requested resource does not exist.	The requested resource does not exist.	Check whether the requested resource exists.
	403	ces.0017	The authentication information is incorrect or the service invoker does not have sufficient rights.	The authentication information is incorrect or the service invoker does not have sufficient rights.	Check whether the user name or password (or AK or SK) or the user rights for obtaining the token are correct.
Cassandra	500	ces.0008	Database error	Database error.	Contact technical support.
Kafka	500	ces.0012	The message queue is abnormal or is not ready.	The message queue is abnormal or is not ready.	Contact technical support.
Zookeeper	500	ces.0021	Internal locking error	Internal locking error	Contact technical support.

Module	HTTP Status Code	Error Code	Error Code Description	Error Message	Measure
Blueflood	500	ces.0019	The metric processing engine is abnormal.	The metric processing engine is abnormal.	Contact technical support.
Alarm	400	ces.0002	The alarm ID cannot be left blank.	The alarm ID must be specified.	Specify the alarm ID.
	403	ces.0018	The number of alarm rules created exceeds the quota.	The number of alarms exceeds the quota	Apply for a higher alarm quota.
	400	ces.0028	The metric and notification type do not match when an alarm rule is created.	The metric does not support the alarm action type.	Modify the metric or notification type according to the parameter description to make them match.

7.3 Obtaining a Project ID

Scenarios

A project ID is required for some URLs when an API is called. Therefore, you need to obtain a project ID in advance. Two methods are available:

- **Obtain the Project ID by Calling an API**
- **Obtain the Project ID from the Console**

Obtain the Project ID by Calling an API

You can obtain the project ID by calling the IAM API used to query project information based on the specified criteria.

The API used to obtain a project ID is GET https://{Endpoint}/v3/projects. {Endpoint} is the IAM endpoint and can be obtained from **Regions and Endpoints**. For details about API authentication, see **Authentication**.

The following is an example response. The value of **id** is the project ID.

```
{
    "projects": [
```



```
        {
            "domain_id": "65382450e8f64ac0870cd180d14e684b",
            "is_domain": false,
            "parent_id": "65382450e8f64ac0870cd180d14e684b",
            "name": "project_name",
            "description": "",
            "links": {
                "next": null,
                "previous": null,
                "self": "https://www.example.com/v3/projects/a4a5d4098fb4474fa22cd05f897d6b99"
            },
            "id": "a4a5d4098fb4474fa22cd05f897d6b99",
            "enabled": true
        }
    ],
    "links": {
        "next": null,
        "previous": null,
        "self": "https://www.example.com/v3/projects"
    }
}
```


Obtain a Project ID from the Console

To obtain a project ID from the console, perform the following operations:

1. Log in to the management console.
2. Click the username and select **My Credentials** from the drop-down list.

   On the **My Credentials** page, view the project ID (value in the **Project ID** column).

A Appendix

A.1 Services Interconnected with Cloud Eye

Category	Service	Namespace	Reference
Compute	Elastic Cloud Server	SYS.ECS	Basic ECS metrics
	ECS (OS monitoring)	AGT.ECS	ECS OS monitoring metrics
	Auto Scaling	SYS.AS	AS metrics
Storage	Elastic Volume Service	SYS.EVS	EVS metrics
	Object Storage Service	SYS.OBS	OBS metrics
	Scalable File Service	SYS.SFS	SFS metrics
Networking	Elastic IP and bandwidth	SYS.VPC	VPC metrics
	Elastic Load Balance	SYS.ELB	ELB metrics
	NAT Gateway	SYS.NAT	NAT Gateway metrics
App Service	Distributed Message Service	SYS.DMS	DMS metrics (Kafka) DMS metrics (RabbitMQ)
	Distributed Cache Service	SYS.DCS	DCS metrics

Category	Service	Namespace	Reference
Database	Relational Database Service	SYS.RDS	RDS for MySQL metrics RDS for PostgreSQL metrics RDS for SQL Server metrics
	Document Database Service	SYS.DDS	DDS metrics

A.2 Events Supported by Event Monitoring

Table A-1 ECS

Event Source	Event Name	Event ID	Event Severity	Description	Solution	Impact
ECS	Recovery started	startAutoRecovery	Major	ECSs on a faulty host would be automatically migrated to another properly-running host. During the migration, the ECSs was restarted.	Wait for the event to end and check whether services are affected.	Services may be interrupted.
	Recovery succeeded	endAutoRecovery	Major	The ECS was recovered after the automatic migration.	This event indicates that the ECS has been recovered and been working properly.	None
	Auto recovery timeout (being processed on the backend)	faultAutoRecovery	Major	Migrating the ECS to a normal host timed out.	Migrate services to other ECSs.	Services are interrupted.

Event Source	Event Name	Event ID	Event Severity	Description	Solution	Impact
	GPU link fault	GPULinkFault	Critical	The GPU of the host running the ECS was faulty or was recovering from a fault.	Deploy service applications in HA mode. After the GPU fault is rectified, check whether services are restored.	Services are interrupted.
	FPGA link fault	FPGALinkFault	Critical	The FPGA of the host running the ECS was faulty or was recovering from a fault.	Deploy service applications in HA mode. After the FPGA fault is rectified, check whether services are restored.	Services are interrupted.
	ECS deleted	deleteServer	Major	The ECS was deleted • on the management console. • by calling APIs.	Check whether the deletion was performed intentionally by a user.	Services are interrupted.

Event Source	Event Name	Event ID	Event Severity	Description	Solution	Impact
	ECS restarted	rebootServer	Minor	The ECS was restarted • on the management console. • by calling APIs.	Check whether the restart was performed intentionally by a user. • Deploy service applications in HA mode. • After the ECS starts up, check whether services recover.	Services are interrupted.
	ECS stopped	stopServer	Minor	The ECS was stopped • on the management console. • by calling APIs. **NOTE** The ECS is stopped only after CTS is enabled. For details, see *Cloud Trace Service User Guide*.	• Check whether the stop operation was performed intentionally by a user. • Deploy service applications in HA mode. • After the ECS starts up, check whether services recover.	Services are interrupted.

Event Source	Event Name	Event ID	Event Severity	Description	Solution	Impact
	NIC deleted	deleteNic	Major	The ECS NIC was deleted • on the management console. • by calling APIs.	• Check whether the deletion was performed intentionally by a user. • Deploy service applications in HA mode. • After the NIC is deleted, check whether services recover.	Services may be interrupted.

Event Source	Event Name	Event ID	Event Severity	Description	Solution	Impact
	ECS resized	resizeServer	Minor	The ECS was resized • on the management console. • by calling APIs.	• Check whether the operation was performed by a user. • Deploy service applications in HA mode. • After the ECS is resized, check whether services have recovered.	Services are interrupted.
	GuestOS restarted	RestartGuestOS	Minor	The guest OS was restarted.	Contact O&M personnel.	Services may be interrupted.
	ECS failure due to abnormal host processes	VMFaultsByHostProcessExceptions	Critical	The processes of the host accommodating the ECS were abnormal.	Contact O&M personnel.	The ECS is faulty.
	Startup failure	faultPowerOn	Major	The ECS failed to start.	Start the ECS again. If the problem persists, contact O&M personnel.	The ECS cannot start.

Event Source	Event Name	Event ID	Event Severity	Description	Solution	Impact
	Host breakdown risk	hostMayCrash	Major	The host where the ECS resides may break down, and the risk cannot be prevented through live migration due to some reasons.	Migrate services running on the ECS first and delete or stop the ECS. Start the ECS only after the O&M personnel eliminate the risk.	The host may break down, causing service interruption.

NOTE

Once a physical host running ECSs breaks down, the ECSs are automatically migrated to a functional physical host. During the migration, the ECSs will be restarted.

Table A-2 EIP

Event Source	Event Name	Event ID	Event Severity
EIP	EIP released	deleteEip	Minor

Table A-3 VPC

Event Source	Event Name	Event ID	Event Severity
VPC	VPC deleted	deleteVpc	Major
	VPC modified	modifyVpc	Minor
	Subnet deleted	deleteSubnet	Minor
	Subnet modified	modifySubnet	Minor
	Bandwidth modified	modifyBandwidth	Minor
	VPN deleted	deleteVpn	Major
	VPN modified	modifyVpn	Minor

Table A-4 EVS

Event Source	Event Name	Event ID	Event Severity	Description	Solution	Impact
EVS	Disk updated	updateVolume	Minor	Update the name and description of an EVS disk.	No further action is required.	None
	Disk expanded	extendVolume	Minor	Expand an EVS disk.	No further action is required.	None
	Disk deleted	deleteVolume	Major	Delete an EVS disk.	No further action is required.	Deleted disks cannot be recovered.
	QoS upper limit reached	reachQoS	Major	The I/O latency increases as the QoS upper limits of the disk are frequently reached and flow control triggered.	Change the disk type to one with a higher specification.	The current disk may fail to meet service requirements.

Event Source	Event Name	Event ID	Event Severity	Description	Solution	Impact
	Faulty storage pool	storagePoolFault	Critical	The faulty storage pool alarm is generated because some data in the pool cannot be accessed.	Contact the EVS personnel to handle this issue. For the resources and services that rely on EVS, determine whether a switchover is required.	Normal read/write operations to disks may be affected. If data cannot be read or written, perform a switchover.

Table A-5 IAM

Event Source	Event Name	Event ID	Event Severity
IAM	Login	login	Minor
	Logout	logout	Minor
	Password changed	changePassword	Major
	User created	createUser	Minor
	User deleted	deleteUser	Major
	User updated	updateUser	Minor
	User group created	createUserGroup	Minor
	User group deleted	deleteUserGroup	Major

Event Source	Event Name	Event ID	Event Severity
	User group updated	updateUserGroup	Minor
	Identity provider created	createIdentityProvider	Minor
	Identity provider deleted	deleteIdentityProvider	Major
	Identity provider updated	updateIdentityProvider	Minor
	Metadata updated	updateMetadata	Minor
	Security policy updated	updateSecurityPolicies	Major
	Credential added	addCredential	Major
	Credential deleted	deleteCredential	Major
	Project created	createProject	Minor
	Project updated	updateProject	Minor
	Project suspended	suspendProject	Major

Table A-6 KMS

Event Source	Event Name	Event ID	Event Severity
KMS	Key disabled	disableKey	Major
	Key deletion scheduled	scheduleKeyDeletion	Minor
	Grant retired	retireGrant	Major
	Grant revoked	revokeGrant	Major

Table A-7 OBS

Event Source	Event Name	Event ID	Event Severity
OBS	Bucket deleted	deleteBucket	Major
	Bucket policy deleted	deleteBucketPolicy	Major

Event Source	Event Name	Event ID	Event Severity
	Bucket ACL configured	setBucketAcl	Minor
	Bucket policy configured	setBucketPolicy	Minor

B Change History

Released On	Description
2022-04-12	This issue is the first official release.

