Application Operations Management

API Reference

Issue 01

Date 2022-06-01

HUAWEI

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.

Contents

1 Before You Start

1.1 Overview

Welcome to *Application Operations Management API Reference*. AOM is a one-stop and multi-dimensional O&M management platform for cloud applications. It monitors your applications and related cloud resources in real time, collects and associates the data of resource metrics, logs, and events to analyze application health status, and provides flexible alarms and abundant data visualization. With AOM, you can detect faults in a timely manner and master the running status of applications, resources, and services in real time.

This document describes how to use application programming interfaces (APIs) to perform operations on AOM, such as creation, deletion, and query. For details about all supported operations, see **API Overview**.

If you plan to call AOM APIs, ensure that you are familiar with AOM concepts.

1.2 API Calling

AOM 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 Concepts

- **Account**

  An account is created upon successful registration with the cloud. 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 and should not be used directly to perform routine management. For security purposes, create users and grant them permissions for routine management.

- **User**

  A user is created in Identity and Access Management (IAM) to use cloud services. Each user has its own identity credentials (password and access key).

  A user can view the account ID and user ID on the **My Credentials** page of the console. The account name, username, and password will be required for API authentication.

- **Region**

  Regions are geographic areas isolated from each other. Resources are region-specific and cannot be used across regions through internal network connections. For low network latency and quick resource access, select the nearest region.

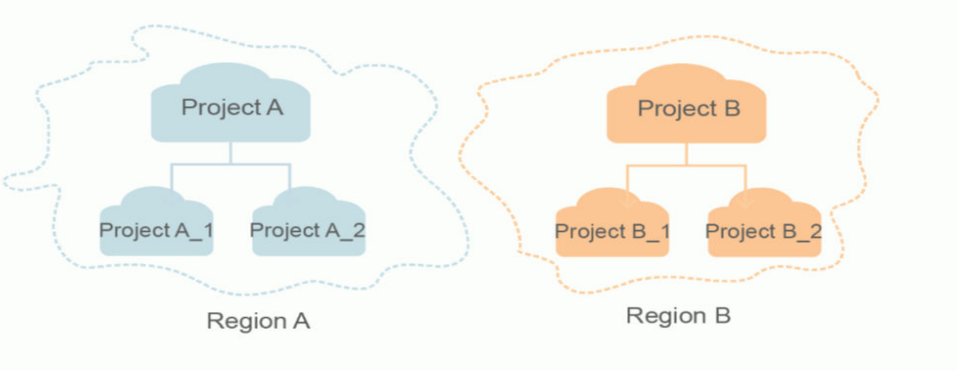
- **AZ**

  AZs are physically isolated locations in a region, but are interconnected through an internal network for enhanced application availability.

- **Project**

  A project corresponds to a region. Projects group and isolate resources (including compute, storage, and network resources) across regions. Users can be granted permissions in a default project to access all resources in the region associated with the project. For more refined access control, create subprojects under a project and create resources in the subprojects. Users can then be assigned permissions to access only specific resources in the subprojects.

Figure 1-1 Project isolating model



2 API Overview

AOM provides open monitoring, auto scaling, and log APIs, helping you quickly implement application O&M with low costs.

Table 2-1 API overview

Type	Description
Monitoring APIs (v1)	Monitoring APIs, including the APIs that query metrics; query and add monitoring data; add, modify, query, and delete threshold rules; add, modify, query, and delete application discovery rules.
Auto Scaling APIs	Auto scaling APIs, including the APIs that create, delete, and update policies; query one or all policies; update and query policy group attributes. **NOTE** Auto scaling APIs do not support CCE clusters of 1.17 or later versions.
Log APIs	Log APIs, including the API that query logs.

Monitoring APIs (v1)

API	Description
Querying Metrics	Query the metrics that can be monitored in the system. You can specify the namespace, metric name, dimension, resource ID (format: **resType_resId**), start position, and maximum number of returned records in pagination queries.
Querying Monitoring Data	Query monitoring data of metrics within a specified time period. You can specify a dimension or period to query.
Adding Monitoring Data	Add one or more monitoring data records.

API	Description
Adding a Threshold Rule	Add a threshold rule.
Modifying a Threshold Rule	Modify a threshold rule.
Querying the Threshold Rule List	Query all threshold rules.
Querying a Threshold Rule	Query a threshold rule.
Deleting a Threshold Rule	Delete a threshold rule.
Adding or Modifying One or More Application Discovery Rules	Add or modify one or more application discovery rules.
Deleting an Application Discovery Rule	Delete one or more application discovery rules.
Querying Application Discovery Rules	Query an application discovery rule.

Auto Scaling APIs

API	Description
Creating a Policy	Create a policy.
Querying the Policy List	Query the details about all policies of a specified project.
Deleting a Policy	Delete a specified policy.
Modifying a Policy	Modify a policy.
Querying a Policy	Query the details about a policy of a specified project.
Modifying Policy Group Attributes	Modify policy group attributes.
Querying Policy Group Attributes	Query policy group attributes.

Log APIs

API	Description
Querying Logs	Query logs by different dimensions, such as the cluster, IP address, or application.

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}

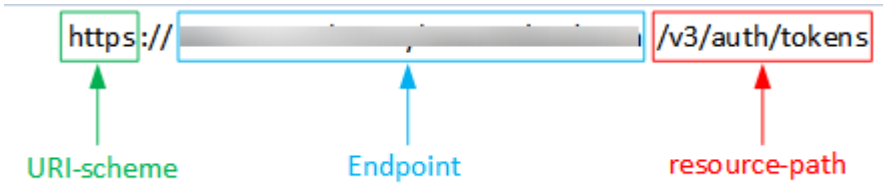
Although a request URI is included in the request header, most programming languages or frameworks require the request URI to be transmitted separately.

- **URI-scheme**: Protocol used to transmit requests. All APIs use HTTPS.
- **Endpoint**: Domain name or IP address of the server where the REST service is deployed. The endpoint varies depending on services and regions.

  For example, the endpoint of IAM in the **ae-ad-1** region is **iam.ae-ad-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**.
- **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 the IAM token in the *XXX* region, obtain the endpoint of IAM for this region and the **resource-path** in the URI of the API used to obtain a user token. Then, construct the URI as follows:

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:

- **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.
- **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://{{endpoint}}/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:

- **Content-Type**: specifies the request body type or format. This field is mandatory and its default value is **application/json**. Other values of this field will be provided for specific APIs if any.
- **X-Auth-Token**: specifies a user token only for token-based API authentication. The user token is a response to the API used to obtain a user token. This API is the only one that does not require authentication.

> **NOTE**
>
> In addition to supporting token-based authentication, APIs also support authentication using access key ID/secret access key (AK/SK). During AK/SK-based authentication, an SDK is used to sign the request, and the **Authorization** (signature information) and **X-Sdk-Date** (time when the request is sent) header fields are automatically added to the request.
>
> For more information, see **AK/SK-based 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://{{endpoint}}/v3/auth/tokens
Content-Type: application/json
```

Request Body

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 ID) with the actual values. To learn how to obtain a project ID, see **Obtaining an Account ID and Project ID**.

> **NOTE**
>
> The **scope** parameter specifies where a token takes effect. You can set **scope** to an account or a project under an account.

```
POST https://Endpoint/v3/auth/tokens
Content-Type: application/json
{
    "auth": {
        "identity": {
            "methods": [
                "password"
            ],
            "password": {
                "user": {
                    "name": "username",
                    "password": "********",
                    "domain": {
                        "name": "domainname"
                    }
                }
            }
        },
        "scope": {
            "project": {
                "id": "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-based authentication: Requests are authenticated using a token.
- Access Key ID/Secret Access Key (AK/SK)-based authentication: Requests are authenticated by encrypting the request body using an AK/SK.

Token-based Authentication

NOTE

The validity period of a token is 24 hours. When using a token for authentication, cache it to prevent frequently calling the Identity and Access Management (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.

In **Making an API Request**, the process of calling the API used to **obtain a user token** is described. After a token is obtained, the **X-Auth-Token** header field must be added to requests to specify the token when other APIs are called. For example, if the token is **ABCDEFJ....**, **X-Auth-Token: ABCDEFJ....** can be added to a request as follows:

```
GET https://Endpoint/v3/auth/projects
Content-Type: application/json
X-Auth-Token: ABCDEFJ....
```

AK/SK-based Authentication

NOTE

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

In AK/SK-based 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 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-based authentication, you can use an AK/SK to sign requests based on the signature algorithm or use the signing SDK to sign requests.

CAUTION

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 the calling of 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

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

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

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

```
{
    "errorCode": "SVCSTG_AMS_4000001",
    "errorMessage": "Request param invalid"
}
```

In the response body, **errorCode** is an error code, and errorMessage provides information about the error.

4 APIs

4.1 Monitoring (v1)

4.1.1 Querying Metrics

Function

This API is used to query the metrics that can be monitored in the system. You can query specific metrics by specifying a namespace, metric name, dimension, and resource ID (format: resType_resId). You can also specify the start position and the maximum number of returned records for a pagination query.

URI

POST /v1/{project_id}/ams/metrics

Table 4-1 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.

Table 4-2 Query Parameters

Parameter	Mandatory	Type	Description
type	No	String	Metric query mode.
limit	No	String	Number of records that can be returned. Value range: 1–1000. Default value: 1000.

Parameter	Mandatory	Type	Description
start	No	String	Start position of a pagination query. The value is a non-negative integer.

Request Parameters

Table 4-3 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM.
Content-Type	Yes	String	Content type, which is application/json. Enumeration values: • **application/json**

Table 4-4 Request body parameters

Parameter	Mandatory	Type	Description
inventoryId	No	String	Resource ID, which must be in the format of resType_resId. Enumerated values of resType: host, application, instance, container, process, network, storage, and volume. When type (a URI parameter) is inventory, this parameter instead of metricItems is used for associated metric queries.
metricItems	No	Array of **QueryMetricItemOptionParam** objects	If type (a URI parameter) is not inventory, the information carried by the array is used to query metrics.

Table 4-5 QueryMetricItemOptionParam

Parameter	Mandatory	Type	Description
dimensions	No	Array of **Dimension** objects	List of metric dimensions.

Parameter	Mandatory	Type	Description
metricName	No	String	Metric name. Length: 1 to 255 characters. Values: cpuUsage, cpuCoreUsed, and other basic metrics provided by AOM. cpuUage: CPU usage. cpuCoreUsed: used CPU cores. Custom metrics.
namespace	Yes	String	Metric namespace. Values: PAAS.CONTAINER: namespace of component, instance, process, and container metrics. PAAS.NODE: namespace of host, network, disk, and file system metrics. PAAS.SLA: namespace of SLA metrics. PAAS.AGGR: namespace of cluster metrics. CUSTOMMETRICS: default namespace of custom metrics. Enumeration values: • **PAAS.CONTAINER** • **PAAS.NODE** • **PAAS.SLA** • **PAAS.AGGR** • **CUSTOMMETRICS**

Table 4-6 Dimension

Parameter	Mandatory	Type	Description
name	Yes	String	Dimension name.
value	Yes	String	Dimension value.

Response Parameters

Status code: 200

Table 4-7 Response body parameters

Parameter	Type	Description
errorCode	String	Response code.
errorMessage	String	Response message.

Parameter	Type	Description
metrics	Array of **MetricItemResultAPI** objects	Metric list.
metaData	**MetaData** object	Metadata, including pagination information.

Table 4-8 MetricItemResultAPI

Parameter	Type	Description
dimensions	Array of **Dimension** objects	List of metric dimensions.
dimensionvaluehash	String	Metric hash value.
metricName	String	Metric name.
namespace	String	Namespace.
unit	String	Metric unit.

Table 4-9 Dimension

Parameter	Type	Description
name	String	Dimension name.
value	String	Dimension value.

Table 4-10 MetaData

Parameter	Type	Description
count	Integer	Number of records that can be returned.
start	String	Start of the next page, which is used for pagination. null: No more data.
total	Integer	Total number of records.

Example Requests

- Query metrics by namespace, appName, and clusterName.


```
POST https://{Endpoint}/v1/{project_id}/ams/metrics

{
  "metricItems" : [ {
    "namespace" : "PAAS.CONTAINER",
    "dimensions" : [ {
      "name" : "appName",
      "value" : "demo"
    }, {
      "name" : "clusterName",
      "value" : "test"
    } ]
  } ]
}
```

- Query metrics by inventory ID.

```
POST https://{Endpoint}/v1/{project_id}/ams/metrics?type=inventory

{
  "inventoryId" : "application_**********7-b56f-fa163e3fee10"
}
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "errorCode" : "SVCSTG_AMS_2000000",
  "errorMessage" : "success",
  "metaData" : {
    "count" : 1,
    "start" : null,
    "total" : 1
  },
  "metrics" : [ {
    "namespace" : "abc",
    "metricName" : "cpuUsage",
    "unit" : "Percent",
    "dimensions" : [ {
      "name" : "instance_id",
      "value" : "demo1"
    } ]
  } ]
}
```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Invalid request. The client should not repeat the request without modifications.
401	Unauthorized: The authorization information is incorrect or invalid.
403	Forbidden: The request is rejected. The server has received the request and understood it, but the server refuses to respond to it. The client should not repeat the request without modifications.

Status Code	Description
500	Internal Server Error: The server is able to receive the request but unable to understand the request.
503	Service Unavailable: The requested service is invalid. The client should not repeat the request without modifications.

Error Codes

See **Error Codes**.

4.1.2 Querying Monitoring Data

Function

This API is used to query monitoring data of metrics within a specified time period. You can specify a dimension or period to query.

URI

POST /v1/{project_id}/ams/metricdata

Table 4-11 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.

Table 4-12 Query Parameters

Parameter	Mandatory	Type	Description
fillValue	No	String	Value filled for breakpoints in monitoring data. Default value: –1. –1: Breakpoints are filled with –1. 0: Breakpoints are filled with 0. null: Breakpoints are filled with null. average: Breakpoints are filled with the average value of the adjacent valid data. If there is no valid data, breakpoints are filled with null. Enumeration values: • **-1** • **0** • **null** • **average**

Request Parameters

Table 4-13 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM.
Content-Type	Yes	String	Content type, which is application/json. Enumeration values: • **application/json**

Table 4-14 Request body parameters

Parameter	Mandatory	Type	Description
metrics	Yes	Array of **MetricQuery MeritcParam** objects	Metric list. Values: A JSON array can contain up to 20 objects.

Parameter	Mandatory	Type	Description
period	Yes	Integer	Granularity for monitoring data. Enumerated value. Values: • 60: 1 minute. • 300: 5 minutes. • 900: 15 minutes. • 3600: 1 hour. Enumeration values: • **60** • **300** • **900** • **3600**
statistics	Yes	Array of strings	Statistic. Values: maximum, minimum, sum, average, and sampleCount.
timerange	Yes	String	Note: Time range/period ≤ 1440 During calculation, the time range and period must be converted to the same unit. Values: Format: start time (UTC, in ms).end time (UTC, in ms).number of minutes in the time period. If both the start time and end time are -1, it indicates the latest N minutes. Time period specified for query. For example, –1.–1.5 indicates the latest 5 minutes. 1501545600000.1501632000000.1440 indicates the fixed time period from 08:00:00 on August 1, 2017 to 08:00:00 on August 2, 2017.

Table 4-15 MetricQueryMeritcParam

Parameter	Mandatory	Type	Description
dimensions	Yes	Array of **Dimension** objects	List of metric dimensions. Note: Neither the array nor the name or value of any dimension in the array can be left blank.

Parameter	Mandatory	Type	Description
metricName	Yes	String	Metric name. Length: 1 to 255 characters. Values: cpuUage: CPU usage. cpuCoreUsed: used CPU cores. Custom metrics.
namespace	Yes	String	Metric namespace. Values: PAAS.CONTAINER: namespace of component, instance, process, and container metrics. PAAS.NODE: namespace of host, network, disk, and file system metrics. PAAS.SLA: namespace of SLA metrics. PAAS.AGGR: namespace of cluster metrics. CUSTOMMETRICS: default namespace of custom metrics.

Table 4-16 Dimension

Parameter	Mandatory	Type	Description
name	Yes	String	Dimension name.
value	Yes	String	Dimension value.

Response Parameters

Status code: 200

Table 4-17 Response body parameters

Parameter	Type	Description
errorCode	String	Response code.
errorMessage	String	Response message.
metrics	Array of **MetricDataValue** objects	Metric list.

Table 4-18 MetricDataValue

Parameter	Type	Description
dataPoints	Array of **MetricDataPoints** objects	Key metric.
metric	**MetricQueryMeritcParam** object	Query parameters.

Table 4-19 MetricDataPoints

Parameter	Type	Description
statistics	Array of **StatisticValue** objects	Statistic.
timestamp	Long	Timestamp.
unit	String	Time series unit.

Table 4-20 StatisticValue

Parameter	Type	Description
statistic	String	Statistic.
value	Double	Statistical result.

Table 4-21 MetricQueryMeritcParam

Parameter	Type	Description
dimensions	Array of **Dimension** objects	List of metric dimensions. Note: Neither the array nor the name or value of any dimension in the array can be left blank.
metricName	String	Metric name. Length: 1 to 255 characters. Values: cpuUage: CPU usage. cpuCoreUsed: used CPU cores. Custom metrics.

Parameter	Type	Description
namespace	String	Metric namespace. Values: PAAS.CONTAINER: namespace of component, instance, process, and container metrics. PAAS.NODE: namespace of host, network, disk, and file system metrics. PAAS.SLA: namespace of SLA metrics. PAAS.AGGR: namespace of cluster metrics. CUSTOMMETRICS: default namespace of custom metrics.

Table 4-22 Dimension

Parameter	Type	Description
name	String	Dimension name.
value	String	Dimension value.

Example Requests

Query the monitoring data of a specified metric in the last five minutes.

```
POST https://{Endpoint}/v1/{project_id}/ams/metricdata

{
  "metrics" : [ {
    "dimensions" : [ {
      "name" : "instance_id",
      "value" : "demo1"
    } ],
    "metricName" : "def",
    "namespace" : "abc"
  } ],
  "period" : 60,
  "statistics" : [ "maximum", "minimum", "sum" ],
  "timerange" : "-1.-1.5"
}
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "errorCode" : "SVCSTG.AMS.2000",
  "errorMessage" : "success",
  "metrics" : [ {
    "metric" : {
      "namespace" : "abc",
      "metricName" : "def",
      "dimensions" : [ {
        "name" : "ghi",
        "value" : "lmn"
      } ]
    },
    "dataPoints" : [ {
```



```
    "timestamp" : "1467892800000",
    "unit" : "Percent",
    "statistics" : [ {
      "statistic" : "maximum",
      "value" : "23"
    } ]
  } ]
 } ]
}
```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Invalid request. The client should not repeat the request without modifications.
401	Unauthorized: The authorization information is incorrect or invalid.
403	Forbidden: The request is rejected. The server has received the request and understood it, but the server refuses to respond to it. The client should not repeat the request without modifications.
500	Internal Server Error: The server is able to receive the request but unable to understand the request.
503	Service Unavailable: The requested service is invalid. The client should not repeat the request without modifications.

Error Codes

See **Error Codes**.

4.1.3 Adding Monitoring Data

Function

This API is used to add one or more monitoring data records to a server.

URI

POST /v1/{project_id}/ams/report/metricdata

Table 4-23 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.

Request Parameters

Table 4-24 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM.
Content-Type	Yes	String	Content type, which is application/json. Enumeration values: • **application/json**

Table 4-25 Request body parameters

Parameter	Mandatory	Type	Description
[items]	Yes	Array of **MetricDataItem** objects	Metric parameters.

Table 4-26 MetricDataItem

Parameter	Mandatory	Type	Description
collect_time	Yes	Long	Data collection time, which ranges from the last 24 hours to the next 0.5 hour. The following requirement must be met: Current UTC time – Data collection time ≤ 24 hours, or Data collection time – Current UTC time ≤ 30 minutes If the data reporting time is earlier than 08:00 of the current day, only the data generated after 08:00 of the current day is displayed on the metric monitoring page. Value range: UNIX timestamp, in ms.
metric	Yes	**RecieveMetricParam** object	Metric details.

Parameter	Mandatory	Type	Description
values	Yes	Array of **RecieveMetricValues** objects	Metric value.

Table 4-27 RecieveMetricParam

Parameter	Mandatory	Type	Description
dimensions	Yes	Array of **Dimension** objects	List of metric dimensions. A maximum of 50 dimensions are supported. Each dimension is in JSON format. The structure is as follows: dimension.name: 1–32 characters. dimension.value: 1–64 characters.
namespace	Yes	String	Metric namespace. The namespace cannot contain any colon (:). It must be in the format of service.item. The value must contain 3 to 32 characters starting with a letter. Only letters, digits, and underscores (_) are allowed. In addition, service cannot be PAAS.

Table 4-28 Dimension

Parameter	Mandatory	Type	Description
name	Yes	String	Dimension name.
value	Yes	String	Dimension value.

Table 4-29 RecieveMetricValues

Parameter	Mandatory	Type	Description
metric_name	Yes	String	Metric name. Length: 1 to 255 characters.

Parameter	Mandatory	Type	Description
type	Yes	String	Data type. Value: int or float. Enumeration values: • **int** • **float**
unit	No	String	Data unit. Length: up to 32 characters.
value	Yes	Double	Metric value, which must be of a valid numeric type.

Response Parameters

Status code: 200

Table 4-30 Response body parameters

Parameter	Type	Description
errorCode	String	Response code.
errorMessage	String	Response message.

Example Requests

Add a monitoring data record to the server. (In the following example, set "collect_time" to the latest timestamp.)

```
POST https://{Endpoint}/v1/{project_id}/ams/report/metricdata

[ {
  "metric" : {
    "namespace" : "NOPAAS.ESC",
    "dimensions" : [ {
      "name" : "instance_id",
      "value" : "instance-101"
    } ]
  },
  "values" : [ {
    "unit" : "percent",
    "metric_name" : "cpu_util",
    "type" : "int",
    "value" : 35
  } ],
  "collect_time" : 1467787152000
} ]
```

Example Responses

Status code: 200

The request is successful.


```
{
  "errorCode" : "SVCSTR.ALS.200200",
  "errorMessage" : "success"
}
```

Status Codes

Status Code	Description
200	The request is successful.
400	The request is invalid.
401	Invalid authentication information.
403	The server has received the request and understood it, but refuse to respond to it.
500	The server is able to receive the request, but the request is improper.
503	The service is unavailable.

Error Codes

See **Error Codes**.

4.1.4 Adding a Threshold Rule

Function

This API is used to add a threshold rule.

URI

POST /v1/{project_id}/ams/alarms

Table 4-31 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.

Request Parameters

Table 4-32 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Project-level token obtained from IAM.
Content-Type	Yes	String	Content type, which is application/json. Enumeration values: • **application/json**

Table 4-33 Request body parameters

Parameter	Mandatory	Type	Description
statistic	Yes	String	Statistic. Enumeration values: • **maximum** • **minimum** • **average** • **sum** • **sampleCount**
namespace	Yes	String	Namespace.
metricName	Yes	String	Metric name. The value must contain 1 to 255 characters long and meet the [a-zA-Z_:] [a-zA-Z0-9_:]* expression. That is, the value must start with a letter, underscore (_), or colon (:). Only letters, digits, underscores, and colons are allowed.
period	Yes	Integer	Statistical period.
alarmLevel	Yes	Integer	Alarm severity.
evaluationPeriods	Yes	Integer	Number of consecutive periods.
comparisonOperator	Yes	String	Threshold criterion expression.
threshold	Yes	String	Threshold.
alarmName	Yes	String	Threshold name.

Parameter	Mandatory	Type	Description
dimensions	Yes	Array of **Dimension** objects	Metric dimension.
unit	Yes	String	Metric unit.
actionEnabled	No	Boolean	Whether to enable alarm reporting.
alarmActions	No	Array of strings	Action to be taken when an alarm is reported.
alarmAdvice	No	String	Suggestion.
alarmDescription	No	String	Threshold rule description.
insufficientDataActions	No	Array of strings	Action to be taken when data is insufficient.
okActions	No	Array of strings	Action to be taken when restoration is complete.

Table 4-34 Dimension

Parameter	Mandatory	Type	Description
name	Yes	String	Dimension name.
value	Yes	String	Dimension value.

Response Parameters

Status code: 200

Table 4-35 Response body parameters

Parameter	Type	Description
errorCode	String	Response code.
errorMessage	String	Response message.
alarmId	Long	Threshold rule code.

Example Requests

Add a threshold rule.

```
POST https://{Endpoint}/v1/{project_id}/ams/alarms
```



```
{
  "actionEnabled" : false,
  "alarmActions" : [ ],
  "alarmAdvice" : "",
  "alarmDescription" : "",
  "alarmLevel" : 3,
  "alarmName" : "aaaaaaaa",
  "comparisonOperator" : ">=",
  "dimensions" : [ {
    "name" : "appName",
    "value" : "rhm-broker"
  } ],
  "evaluationPeriods" : 1,
  "insufficientDataActions" : [ ],
  "metricName" : "cpuCoreLimit",
  "namespace" : "PAAS.CONTAINER",
  "okActions" : [ ],
  "period" : 60000,
  "statistic" : "average",
  "threshold" : 0,
  "unit" : "Core"
}
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "errorCode" : "SVCSTG_AMS_2000000",
  "errorMessage" : "success",
  "alarmId" : 12345678
}
```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Invalid request. The client should not repeat the request without modifications.
401	Unauthorized: The authorization information is incorrect or invalid.
403	Forbidden: The request is rejected. The server has received the request and understood it, but the server refuses to respond to it. The client should not repeat the request without modifications.
500	Internal Server Error: The server is able to receive the request but unable to understand the request.
503	Service Unavailable: The requested service is invalid. The client should not repeat the request without modifications.

Error Codes

See **Error Codes**.

4.1.5 Modifying a Threshold Rule

Function

This API is used to modify a threshold rule.

URI

PUT /v1/{project_id}/ams/alarms

Table 4-36 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.

Request Parameters

Table 4-37 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Project-level token obtained from IAM.
Content-Type	Yes	String	Content type, which is application/json. Enumeration values: • **application/json**

Table 4-38 Request body parameters

Parameter	Mandatory	Type	Description
statistic	Yes	String	Statistic. Enumeration values: • **maximum** • **minimum** • **average** • **sum** • **sampleCount**
namespace	Yes	String	Namespace.

Parameter	Mandatory	Type	Description
metricName	Yes	String	Metric name. The value must contain 1 to 255 characters long and meet the [a-zA-Z_:] [a-zA-Z0-9_:]* expression. That is, the value must start with a letter, underscore (_), or colon (:). Only letters, digits, underscores, and colons are allowed.
period	Yes	Integer	Statistical period.
alarmLevel	Yes	Integer	Alarm severity.
evaluationPeriods	Yes	Integer	Number of consecutive periods.
comparisonOperator	Yes	String	Threshold criterion expression.
threshold	Yes	String	Threshold.
alarmName	Yes	String	Threshold name.
dimensions	Yes	Array of **Dimension** objects	Metric dimension.
unit	Yes	String	Metric unit.
actionEnabled	No	Boolean	Whether to enable alarm reporting.
alarmActions	No	Array of strings	Action to be taken when an alarm is reported.
alarmAdvice	No	String	Suggestion.
alarmDescription	No	String	Threshold rule description.
insufficientDataActions	No	Array of strings	Action to be taken when data is insufficient.
okActions	No	Array of strings	Action to be taken when restoration is complete.

Table 4-39 Dimension

Parameter	Mandatory	Type	Description
name	Yes	String	Dimension name.

Parameter	Mandatory	Type	Description
value	Yes	String	Dimension value.

Response Parameters

Status code: 200

Table 4-40 Response body parameters

Parameter	Type	Description
errorCode	String	Response code.
errorMessage	String	Response message.
alarmId	Long	Threshold rule code.

Example Requests

Modify a threshold rule.

```
PUT https://{Endpoint}/v1/{project_id}/ams/alarms

{
  "actionEnabled" : false,
  "alarmActions" : [ ],
  "alarmAdvice" : "",
  "alarmDescription" : "",
  "alarmLevel" : 3,
  "alarmName" : "aaaaaaaa",
  "comparisonOperator" : ">=",
  "dimensions" : [ {
    "name" : "appName",
    "value" : "rhm-broker"
  } ],
  "evaluationPeriods" : 1,
  "insufficientDataActions" : [ ],
  "metricName" : "cpuCoreLimit",
  "namespace" : "PAAS.CONTAINER",
  "okActions" : [ ],
  "period" : 60000,
  "statistic" : "average",
  "threshold" : 0,
  "unit" : "Core"
}
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "errorCode" : "SVCSTG_AMS_2000000",
  "errorMessage" : "success",
  "alarmId" : 12345678
}
```


Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Invalid request. The client should not repeat the request without modifications.
401	Unauthorized: The authorization information is incorrect or invalid.
403	Forbidden: The request is rejected. The server has received the request and understood it, but the server refuses to respond to it. The client should not repeat the request without modifications.
500	Internal Server Error: The server is able to receive the request but unable to understand the request.
503	Service Unavailable: The requested service is invalid. The client should not repeat the request without modifications.

Error Codes

See **Error Codes**.

4.1.6 Querying the Threshold Rule List

Function

This API is used to query the threshold rule list.

URI

GET /v1/{project_id}/ams/alarms

Table 4-41 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.

Table 4-42 Query Parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Value range: 1–1000. Default value: 1000. Number of records that can be returned.

Parameter	Mandatory	Type	Description
start	No	Long	Pagination information.

Request Parameters

Table 4-43 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Project-level token obtained from IAM.
Content-Type	Yes	String	Content type, which is application/json. Enumeration values: • **application/json**

Response Parameters

Status code: 200

Table 4-44 Response body parameters

Parameter	Type	Description
errorCode	String	Response code.
errorMessage	String	Response message.
metaData	**MetaData** object	Metadata, including pagination information.
thresholds	Array of **AlarmAPIQueryAlarmResult** objects	Threshold rule list.

Table 4-45 MetaData

Parameter	Type	Description
count	Integer	Number of records that can be returned.
start	String	Start of the next page, which is used for pagination. null: No more data.
total	Integer	Total number of records.

Table 4-46 AlarmAPIQueryAlarmResult

Parameter	Type	Description
idTurnOn	Boolean	Whether to enable the threshold rule.
type	String	Threshold rule type.
policyName	String	Threshold rule template name.
alarmName	String	Threshold rule name.
id	String	Threshold rule ID.
alarmDescription	String	Threshold rule description.
actionEnabled	Boolean	Whether to enable notification.
okActions	Array of strings	Action to be taken when restoration is complete.
alarmActions	Array of strings	Action to be taken when an alarm is reported.
insufficientDataActions	Array of strings	Action to be taken when data is insufficient.
stateValue	String	Service status.
stateReason	String	Cause description.
stateUpdatedTimestamp	String	Time when the status was updated.
metricName	String	Time series name.
namespace	String	Namespace of time series objects.
statistic	String	Statistic.
dimensions	Array of **Dimension** objects	List of time series dimensions.
resources	Array of strings	Resource information (discarded).
period	Integer	Statistical period.
evaluationPeriods	Integer	Number of consecutive periods.
unit	String	Threshold unit.
threshold	String	Threshold value.
comparisonOperator	String	Comparison operator.

Parameter	Type	Description
alarmAdvice	String	Alarm clearance suggestion.
alarmLevel	String	Alarm severity.

Table 4-47 Dimension

Parameter	Type	Description
name	String	Dimension name.
value	String	Dimension value.

Example Requests

Query the threshold rule list.

```
GET https://{Endpoint}/v1/{project_id}/ams/alarms
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "errorCode" : "SVCSTG_AMS_2000000",
  "errorMessage" : "success",
  "metaData" : {
    "count" : 10,
    "start" : null,
    "total" : 100
  },
  "thresholds" : [ {
    "id" : "2137",
    "alarmName" : "aaaaaaaa",
    "alarmDescription" : "",
    "actionEnabled" : false,
    "okActions" : [ ],
    "alarmActions" : [ ],
    "insufficientDataActions" : [ ],
    "stateValue" : "alarm",
    "stateReason" : "",
    "stateUpdatedTimestamp" : null,
    "metricName" : "cpuCoreLimit",
    "namespace" : "PAAS.CONTAINER",
    "statistic" : "average",
    "dimensions" : [ {
      "name" : "appName",
      "value" : "rhm-broker"
    } ],
    "period" : 60000,
    "evaluationPeriods" : 1,
    "unit" : "Core",
    "threshold" : "0",
    "comparisonOperator" : ">=",
    "alarmAdvice" : "",
    "alarmLevel" : 3
```



```
 }]
}
```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Invalid request. The client should not repeat the request without modifications.
401	Unauthorized: The authorization information is incorrect or invalid.
403	Forbidden: The request is rejected. The server has received the request and understood it, but the server refuses to respond to it. The client should not repeat the request without modifications.
500	Internal Server Error: The server is able to receive the request but unable to understand the request.
503	Service Unavailable: The requested service is invalid. The client should not repeat the request without modifications.

Error Codes

See **Error Codes**.

4.1.7 Querying a Threshold Rule

Function

This API is used to query a threshold rule.

URI

GET /v1/{project_id}/ams/alarms/{alarm_id}

Table 4-48 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.
alarm_id	Yes	String	Threshold rule ID.

Request Parameters

Table 4-49 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Project-level token obtained from IAM.
Content-Type	Yes	String	Content type, which is application/json. Enumeration values: • **application/json**

Response Parameters

Status code: 200

Table 4-50 Response body parameters

Parameter	Type	Description
errorCode	String	Response code.
errorMessage	String	Response message.
metaData	**MetaData** object	Metadata, including pagination information.
thresholds	Array of **AlarmAPIQueryAlarmResult** objects	Threshold rule list.

Table 4-51 MetaData

Parameter	Type	Description
count	Integer	Number of records that can be returned.
start	String	Start of the next page, which is used for pagination. null: No more data.
total	Integer	Total number of records.

Table 4-52 AlarmAPIQueryAlarmResult

Parameter	Type	Description
idTurnOn	Boolean	Whether to enable the threshold rule.
type	String	Threshold rule type.
policyName	String	Threshold rule template name.
alarmName	String	Threshold rule name.
id	String	Threshold rule ID.
alarmDescription	String	Threshold rule description.
actionEnabled	Boolean	Whether to enable notification.
okActions	Array of strings	Action to be taken when restoration is complete.
alarmActions	Array of strings	Action to be taken when an alarm is reported.
insufficientDataActions	Array of strings	Action to be taken when data is insufficient.
stateValue	String	Service status.
stateReason	String	Cause description.
stateUpdatedTimestamp	String	Time when the status was updated.
metricName	String	Time series name.
namespace	String	Namespace of time series objects.
statistic	String	Statistic.
dimensions	Array of **Dimension** objects	List of time series dimensions.
resources	Array of strings	Resource information (discarded).
period	Integer	Statistical period.
evaluationPeriods	Integer	Number of consecutive periods.
unit	String	Threshold unit.
threshold	String	Threshold value.
comparisonOperator	String	Comparison operator.

Parameter	Type	Description
alarmAdvice	String	Alarm clearance suggestion.
alarmLevel	String	Alarm severity.

Table 4-53 Dimension

Parameter	Type	Description
name	String	Dimension name.
value	String	Dimension value.

Example Requests

Query a threshold rule.

```
GET https://{Endpoint}/v1/{project_id}/ams/alarms/{alarm_id}
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "errorCode" : "SVCSTG_AMS_2000000",
  "errorMessage" : "success",
  "metaData" : {
    "count" : 10,
    "start" : null,
    "total" : 100
  },
  "thresholds" : [ {
    "id" : "2137",
    "alarmName" : "aaaaaaaa",
    "alarmDescription" : "",
    "actionEnabled" : false,
    "okActions" : [ ],
    "alarmActions" : [ ],
    "insufficientDataActions" : [ ],
    "stateValue" : "alarm",
    "stateReason" : "",
    "stateUpdatedTimestamp" : null,
    "metricName" : "cpuCoreLimit",
    "namespace" : "PAAS.CONTAINER",
    "statistic" : "average",
    "dimensions" : [ {
      "name" : "appName",
      "value" : "rhm-broker"
    } ],
    "period" : 60000,
    "evaluationPeriods" : 1,
    "unit" : "Core",
    "threshold" : "0",
    "comparisonOperator" : ">=",
    "alarmAdvice" : "",
    "alarmLevel" : 3
```



```
 }]
}
```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Invalid request. The client should not repeat the request without modifications.
401	Unauthorized: The authorization information is incorrect or invalid.
403	Forbidden: The request is rejected. The server has received the request and understood it, but the server refuses to respond to it. The client should not repeat the request without modifications.
500	Internal Server Error: The server is able to receive the request but unable to understand the request.
503	Service Unavailable: The requested service is invalid. The client should not repeat the request without modifications.

Error Codes

See **Error Codes**.

4.1.8 Deleting a Threshold Rule

Function

This API is used to delete a threshold rule.

URI

DELETE /v1/{project_id}/ams/alarms/{alarm_id}

Table 4-54 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.
alarm_id	Yes	String	Threshold rule ID.

Request Parameters

Table 4-55 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Project-level token obtained from IAM.
Content-Type	Yes	String	Content type, which is application/json. Enumeration values: • **application/json**

Response Parameters

Status code: 200

Table 4-56 Response body parameters

Parameter	Type	Description
errorCode	String	Response code.
errorMessage	String	Response message.

Example Requests

Delete a threshold rule.

```
DELETE https://{Endpoint}/v1/{project_id}/ams/alarms/{alarm_id}
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "errorCode" : "SVCSTG_AMS_2000000",
  "errorMessage" : "Delete Threshold [aaaaaaaa] successfully"
}
```

Status Codes

Status Code	Description
200	OK: The request is successful.

Status Code	Description
400	Bad Request: Invalid request. The client should not repeat the request without modifications.
401	Unauthorized: The authorization information is incorrect or invalid.
403	Forbidden: The request is rejected. The server has received the request and understood it, but the server refuses to respond to it. The client should not repeat the request without modifications.
500	Internal Server Error: The server is able to receive the request but unable to understand the request.
503	Service Unavailable: The requested service is invalid. The client should not repeat the request without modifications.

Error Codes

See **Error Codes**.

4.1.9 Adding or Modifying One or More Application Discovery Rules

Function

This API is used to add or modify one or more application discovery rules. A maximum of 100 rules can be added to a project.

URI

PUT /v1/{project_id}/inv/servicediscoveryrules

Table 4-57 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.

Request Parameters

Table 4-58 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM.
Content-Type	Yes	String	Content type, which is application/json.

Table 4-59 Request body parameters

Parameter	Mandatory	Type	Description
appRules	No	Array of **AppRules** objects	Service parameters.

Table 4-60 AppRules

Parameter	Mandatory	Type	Description
createTime	No	String	Creation time. When creating an application discovery rule, leave this parameter blank. When modifying an application discovery rule, enter the returned createTime.
enable	Yes	Boolean	Whether a rule is enabled. Value: true or false.
eventName	Yes	String	aom_inventory_rules_event Rule event name. For application discovery, the fixed value is aom_inventory_rules_event. Enumeration values: • **aom_inventory_rules_event**
hostid	No	Array of strings	Host ID (not used currently and can be left empty).

Parameter	Mandatory	Type	Description
id	Yes	String	Rule ID. When creating an application discovery rule, leave this parameter blank. When modifying an application discovery rule, enter a rule ID.
name	Yes	String	Rule name. The value can contain a maximum of 64 characters. It must start with a lowercase letter and cannot end with a hyphen (-). Only digits, lowercase letters, and hyphens are allowed.
projectid	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.
spec	Yes	**AppRulesSpec** object	Rule details.

Table 4-61 AppRulesSpec

Parameter	Mandatory	Type	Description
appType	No	String	Service type, which is used only for rule classification and UI display. You can enter any field. For example, enter Java or Python by technology stack. You can also enter collector or database by function.
attrList	No	Array of strings	Attribute list (not used currently and can be left empty). Value: cmdLine or env.
detectLog	No	String	Whether to enable log collection. Value: true or false.

Parameter	Mandatory	Type	Description
discoveryRule	Yes	Array of **DiscoveryRule** objects	Discovery rule. When it is an array consisting of multiple conditions, only the processes that meet all the conditions are filtered. If the value of checkType is cmdLine, set the value of checkMode to contain. checkContent is in the format of ["xxx"], indicating that the process must contain the xxx parameter. If the value of checkType is env, set the value of checkMode to contain. checkContent is in the format of ["k1","v1"], indicating that the process must contain the environment variable whose name is k1 and value is v1. If the value of checkType is scope, set the value of checkMode to equals. checkContent is in the format of ["hostId1","hostId2"], indicating that the rule takes effect only on specified nodes. If no nodes are specified, the rule applies to all nodes of the project.
isDefaultRule	Yes	String	Whether the current rule is the default one. Value: true or false.
isDetect	Yes	String	Whether the scenario is a pre-check scenario. No rules will be saved in the pre-check scenario. This scenario is designed only to check whether a rule can detect node processes before it is delivered. Value: true or false.
logFileFix	No	Array of strings	Log file suffix. Value: log, trace, or out.

Parameter	Mandatory	Type	Description
logPathRule	No	Array of **LogPathRule** objects	Log path configuration rule. If cmdLineHash is a fixed string, a log path or log file is specified. Otherwise, only the files whose names end with .log and .trace are collected. If the value of nameType is cmdLineHash, args is in the format of ["00001"] and value is in the format of ["/xxx/xx.log"], indicating that the log path is /xxx/xx.log when the startup command is 00001.
nameRule	Yes	**NameRule** object	Naming rules for discovered services and applications.
priority	Yes	String	Rule priority. An integer ranging from 1 to 9999. The default value is 9999.

Table 4-62 DiscoveryRule

Parameter	Mandatory	Type	Description
checkContent	Yes	Array of strings	Matched value.
checkMode	Yes	String	Match condition. Value: contain or equals.
checkType	Yes	String	Match type. Value: cmdLine, env, or scope.

Table 4-63 LogPathRule

Parameter	Mandatory	Type	Description
args	No	Array of strings	Command.
nameType	No	String	Value type. Option: cmdLineHash.
value	No	Array of strings	Log path.

Table 4-64 NameRule

Parameter	Mandatory	Type	Description
appNameRule	Yes	Array of **AppNameRule** objects	Service name rule. If there are multiple objects in the array, the character strings extracted from these objects constitute the service name. If the value of nameType is cmdLine, args is in the format of ["start", "end"], indicating that the characters between start and end in the command are extracted. If the value of nameType is cmdLine, args is in the format of ["aa"], indicating that the environment variable named aa is extracted. If the value of nameType is str, args is in the format of ["fix"], indicating that the service name is suffixed with fix. If the value of nameType is cmdLineHash, args is in the format of ["0001"] and value is in the format of ["ser"], indicating that the service name is ser when the startup command is 0001.

Parameter	Mandatory	Type	Description
applicationNameRule	Yes	Array of **ApplicationNameRule** objects	Application name rule. If the value of nameType is cmdLine, args is in the format of ["start", "end"], indicating that the characters between start and end in the command are extracted. If the value of nameType is cmdLine, args is in the format of ["aa"], indicating that the environment variable named aa is extracted. If the value of nameType is str, args is in the format of ["fix"], indicating that the service name is suffixed with fix. If the value of nameType is cmdLineHash, args is in the format of ["0001"] and value is in the format of ["ser"], indicating that the application name is ser when the startup command is 0001.

Table 4-65 AppNameRule

Parameter	Mandatory	Type	Description
nameType	Yes	String	Value type. Options: cmdLineHash, cmdLine, env, and str.
args	Yes	Array of strings	Input value.
value	No	Array of strings	Service name, which is mandatory only if the value of nameType is cmdLineHash.

Table 4-66 ApplicationNameRule

Parameter	Mandatory	Type	Description
nameType	Yes	String	Value type. Options: cmdLineHash, cmdLine, env, and str.

Parameter	Mandatory	Type	Description
args	Yes	Array of strings	Input value.
value	No	Array of strings	Service name, which is mandatory only if the value of nameType is cmdLineHash.

Response Parameters

Status code: 200

Table 4-67 Response body parameters

Parameter	Type	Description
errorCode	String	Response code.
errorMessage	String	Response message.
responseStatus	Integer	Response status code.

Example Requests

Add or modify one or more application discovery rules.

```
PUT https://{Endpoint}/v1/{project_id}/inv/servicediscoveryrules

{
  "appRules" : [ {
    "id" : "44d6c4bb-f673-4bf4-8d33-313832f37b28",
    "name" : "bytest",
    "createTime" : "",
    "projectid" : "5a6036f48e954fcd84d198cb28db311a",
    "enable" : true,
    "hostid" : [ ],
    "eventName" : "aom_inventory_rules_event",
    "spec" : {
      "detectLog" : "true",
      "logFileFix" : [ "log", "trace" ],
      "discoveryRule" : [ {
        "checkType" : "cmdLine",
        "checkMode" : "contain",
        "checkContent" : [ "default" ]
      }, {
        "checkType" : "scope",
        "checkMode" : "equals",
        "checkContent" : [ "44d6c4bb-f673-4bf4-8d33-313832f37b28" ]
      } ],
      "attrList" : [ "cmdLine" ],
      "isDetect" : "false",
      "priority" : "1",
      "nameRule" : {
        "appNameRule" : [ {
          "nameType" : "cmdLineHash",
          "args" : [ "00000000001" ],
```



```
        "value" : [ "serviceName1" ]
      }, {
        "nameType" : "cmdLine",
        "args" : [ "/var/paas/kubernetes/", "/kubeconfig" ]
      }, {
        "nameType" : "env",
        "args" : [ "APP_NAME" ]
      }, {
        "nameType" : "str",
        "args" : [ "kube" ]
      } ],
      "applicationNameRule" : [ {
        "nameType" : "cmdLineHash",
        "args" : [ "00000000001" ],
        "value" : [ "applicationName1" ]
      }, {
        "nameType" : "str",
        "args" : [ "kubeproxy" ]
      } ]
    },
    "appType" : "",
    "isDefaultRule" : "false",
    "logPathRule" : [ {
      "nameType" : "cmdLineHash",
      "args" : [ "00000000001" ],
      "value" : [ "/xx/xxx/xx.log", "/xx/xxx/xx" ]
    } ]
  }
 } ]
}
```

Example Responses

Status code: 200

OK The request is successful.

```
{
  "errorCode" : "SVCSTG.INV.2000000",
  "errorMessage" : "success",
  "id" : [ "44d6c4bb-f673-4bf4-8d33-313832f37b28" ]
}
```

Status Codes

Status Code	Description
200	OK The request is successful.
400	Bad Request Invalid request. The client should not repeat the request without modifications.
401	Unauthorized The authentication information is incorrect or invalid.
403	Forbidden The request is rejected. The server has received the request and understood it, but the server refuses to respond to it. The client should not repeat the request without modifications.
500	Internal Server Error The server is able to receive the request but unable to understand the request.

Status Code	Description
503	Service Unavailable The requested service is invalid. The client should not repeat the request without modifications.

Error Codes

See **Error Codes**.

4.1.10 Deleting an Application Discovery Rule

Function

This API is used to delete an application discovery rule.

URI

DELETE /v1/{project_id}/inv/servicediscoveryrules

Table 4-68 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.

Table 4-69 Query Parameters

Parameter	Mandatory	Type	Description
appRulesIds	Yes	Array	Discovery rule ID. IDs need to be separated by commas (,).

Request Parameters

Table 4-70 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM.
Content-Type	Yes	String	Content type, which is application/json.

Response Parameters

Status code: 200

Table 4-71 Response body parameters

Parameter	Type	Description
errorCode	String	Response code.
errorMessage	String	Response message.
responseStatus	Integer	Response status code.

Example Requests

Delete an application discovery rule with a specified ID.

```
DELETE https://{Endpoint}/v1/{project_id}/inv/servicediscoveryrules?appRulesIds=b788349e-62b2-3c7a-b597-02c611d59801
```

Example Responses

Status code: 200

OK The request is successful.

```
{
  "errorCode" : "SVCSTG.INV.2000000",
  "errorMessage" : null
}
```

Status Codes

Status Code	Description
200	OK The request is successful.
400	Bad Request Invalid request. The client should not repeat the request without modifications.
401	Unauthorized The authorization information is incorrect or invalid.
403	The server has received the request and understood it, but refuse to respond to it. The server has received the request and understood it, but the server refuses to respond to it. The client should not repeat the request without modifications.
500	Internal Server Error The server is able to receive the request but unable to understand the request.
503	Service Unavailable The requested service is invalid. The client should not repeat the request without modifications.

Error Codes

See **Error Codes**.

4.1.11 Querying Application Discovery Rules

Function

This API is used to query existing application discovery rules in the system.

URI

GET /v1/{project_id}/inv/servicediscoveryrules

Table 4-72 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.

Table 4-73 Query Parameters

Parameter	Mandatory	Type	Description
id	No	String	Application discovery rule ID, which corresponds to an application discovery rule. If this parameter is left blank, all application discovery rules in the project are returned.

Request Parameters

Table 4-74 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM.
Content-Type	Yes	String	Content type, which is application/json.

Response Parameters

Status code: 200

Table 4-75 Response body parameters

Parameter	Type	Description
appRules	Array of **AppRules** objects	Rule information.
errorCode	String	Response code. AOM_INVENTORY_2000000: Success response.
errorMessage	String	Response message.

Table 4-76 AppRules

Parameter	Type	Description
createTime	String	Creation time. When creating an application discovery rule, leave this parameter blank. When modifying an application discovery rule, enter the returned createTime.
enable	Boolean	Whether a rule is enabled. Value: true or false.
eventName	String	aom_inventory_rules_event Rule event name. For application discovery, the fixed value is aom_inventory_rules_event. Enumeration values: • **aom_inventory_rules_event**
hostid	Array of strings	Host ID (not used currently and can be left empty).
id	String	Rule ID. When creating an application discovery rule, leave this parameter blank. When modifying an application discovery rule, enter a rule ID.
name	String	Rule name. The value can contain a maximum of 64 characters. It must start with a lowercase letter and cannot end with a hyphen (-). Only digits, lowercase letters, and hyphens are allowed.
projectid	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.
spec	**AppRulesSpec** object	Rule details.

Table 4-77 AppRulesSpec

Parameter	Type	Description
appType	String	Service type, which is used only for rule classification and UI display. You can enter any field. For example, enter Java or Python by technology stack. You can also enter collector or database by function.
attrList	Array of strings	Attribute list (not used currently and can be left empty). Value: cmdLine or env.
detectLog	String	Whether to enable log collection. Value: true or false.
discoveryRule	Array of **DiscoveryRule** objects	Discovery rule. When it is an array consisting of multiple conditions, only the processes that meet all the conditions are filtered. If the value of checkType is cmdLine, set the value of checkMode to contain. checkContent is in the format of ["xxx"], indicating that the process must contain the xxx parameter. If the value of checkType is env, set the value of checkMode to contain. checkContent is in the format of ["k1","v1"], indicating that the process must contain the environment variable whose name is k1 and value is v1. If the value of checkType is scope, set the value of checkMode to equals. checkContent is in the format of ["hostId1","hostId2"], indicating that the rule takes effect only on specified nodes. If no nodes are specified, the rule applies to all nodes of the project.
isDefaultRule	String	Whether the current rule is the default one. Value: true or false.
isDetect	String	Whether the scenario is a pre-check scenario. No rules will be saved in the pre-check scenario. This scenario is designed only to check whether a rule can detect node processes before it is delivered. Value: true or false.
logFileFix	Array of strings	Log file suffix. Value: log, trace, or out.

Parameter	Type	Description
logPathRule	Array of **LogPathRule** objects	Log path configuration rule. If cmdLineHash is a fixed string, a log path or log file is specified. Otherwise, only the files whose names end with .log and .trace are collected. If the value of nameType is cmdLineHash, args is in the format of ["00001"] and value is in the format of ["/xxx/xx.log"], indicating that the log path is /xxx/xx.log when the startup command is 00001.
nameRule	**NameRule** object	Naming rules for discovered services and applications.
priority	String	Rule priority. An integer ranging from 1 to 9999. The default value is 9999.

Table 4-78 DiscoveryRule

Parameter	Type	Description
checkContent	Array of strings	Matched value.
checkMode	String	Match condition. Value: contain or equals.
checkType	String	Match type. Value: cmdLine, env, or scope.

Table 4-79 LogPathRule

Parameter	Type	Description
args	Array of strings	Command.
nameType	String	Value type. Option: cmdLineHash.
value	Array of strings	Log path.

Table 4-80 NameRule

Parameter	Type	Description
appNameRule	Array of **AppNameRule** objects	Service name rule. If there are multiple objects in the array, the character strings extracted from these objects constitute the service name. If the value of nameType is cmdLine, args is in the format of ["start", "end"], indicating that the characters between start and end in the command are extracted. If the value of nameType is cmdLine, args is in the format of ["aa"], indicating that the environment variable named aa is extracted. If the value of nameType is str, args is in the format of ["fix"], indicating that the service name is suffixed with fix. If the value of nameType is cmdLineHash, args is in the format of ["0001"] and value is in the format of ["ser"], indicating that the service name is ser when the startup command is 0001.
applicationNameRule	Array of **ApplicationNameRule** objects	Application name rule. If the value of nameType is cmdLine, args is in the format of ["start", "end"], indicating that the characters between start and end in the command are extracted. If the value of nameType is cmdLine, args is in the format of ["aa"], indicating that the environment variable named aa is extracted. If the value of nameType is str, args is in the format of ["fix"], indicating that the service name is suffixed with fix. If the value of nameType is cmdLineHash, args is in the format of ["0001"] and value is in the format of ["ser"], indicating that the application name is ser when the startup command is 0001.

Table 4-81 AppNameRule

Parameter	Type	Description
nameType	String	Value type. Options: cmdLineHash, cmdLine, env, and str.
args	Array of strings	Input value.
value	Array of strings	Service name, which is mandatory only if the value of nameType is cmdLineHash.

Table 4-82 ApplicationNameRule

Parameter	Type	Description
nameType	String	Value type. Options: cmdLineHash, cmdLine, env, and str.
args	Array of strings	Input value.
value	Array of strings	Service name, which is mandatory only if the value of nameType is cmdLineHash.

Example Requests

None

Example Responses

Status code: 200

OK The request is successful.


```
{
  "appRules" : [ {
    "createTime" : "1599098476654",
    "enable" : true,
    "name" : "ica**nt",
    "eventName" : "aom_in***tory_rules_event",
    "hostid" : [ ],
    "id" : "b53a5152-****-****-****-302367e04c0b",
    "projectid" : "2a473356c*********be891bffc1cf",
    "spec" : {
      "detectLog" : "true",
      "editable" : null,
      "logPathRule" : [ ],
      "priority" : 9999,
      "attrList" : [ "cmdLine" ],
      "nameRule" : {
        "appNameRule" : [ {
          "nameType" : "cmdLineHash",
          "args" : [ "/opt/******** -DNFW=ica**nt" ],
          "value" : [ "aicagentserver" ]
        } ],
        "applicationNameRule" : [ {
          "nameType" : "cmdLineHash",
          "args" : [ "/opt/********** -DNFW=ica**nt" ],
          "value" : [ "aica**nt" ]
        } ]
      },
      "appType" : "",
      "aom_metric_relabel_configs" : null,
      "logFileFix" : [ "log", "trace", "out" ],
      "isDetect" : "false",
      "isDefaultRule" : null,
      "dataSource" : null,
      "discoveryRule" : [ {
        "checkType" : "cmdLine",
        "checkContent" : [ "-DNFW=ica**nt" ],
        "checkMode" : "contain"
      } ]
    },
    "desc" : null
```



```
 } ],
 "errorMessage" : null,
 "errorCode" : "SVCSTG.INV.2000000",
 "responseStatus" : 200
}
```

Status Codes

Status Code	Description
200	OK The request is successful.
400	Bad Request Invalid request. The client should not repeat the request without modifications.
401	Unauthorized The authentication information is incorrect or invalid.
403	Forbidden The request is rejected. The server has received the request and understood it, but the server refuses to respond to it. The client should not repeat the request without modifications.
500	Internal Server Error The server is able to receive the request but unable to understand the request.
503	Service Unavailable The requested service is invalid. The client should not repeat the request without modifications.

Error Codes

See **Error Codes**.

4.2 Auto Scaling

4.2.1 Creating a Policy

Function

This API is used to create a policy. [It is not supported in versions later than 1.8.0.] (tag:hcs) The value must contain 1 to 64 characters starting with a letter. Only digits, letters, underscores (_), and hyphens (-) are allowed. In an AS group, for the same metric (metric_name), the value of metric_threshold with metric_operation set to > must be greater than that of metric_threshold with metric_operation set to <. In an AS group, you can create only one alarm policy with the same metric_operation for each metric. In a policy, conditions of metrics with the same metric_name cannot conflict. The year in the trigger time (launch_time) of a scheduled policy cannot be later than 2099. The year in the start time (start_time) and end time (end_time) of a periodic policy cannot be later than 2099. An AS group supports a maximum of 10 scheduled and periodic policies, and 10 alarm policies. In an AS group, alarm policies cannot affect each other.

URI

POST /v1/{project_id}/pe/policy

Table 4-83 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID

Request Parameters

Table 4-84 Request header parameters

Parameter	Mandatory	Type	Description
resourcetype	Yes	String	Resource type. Example: node
Content-Type	Yes	String	Content type, which is application/json;charset=utf-8. Enumeration values: • **application/ json;charset=utf-8**
Cluster-Id	Yes	String	Cluster ID.
Namespace	Yes	String	Namespace.
X-Auth-Token	Yes	String	User token obtained from IAM.
Reserved-Info	No	String	Custom field.

Table 4-85 Request body parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Policy name.
policy_type	Yes	String	Policy type. Value: SCHEDULED RECURRENCE ALARM Enumeration values: • **SCHEDULED** • **RECURRENCE** • **ALARM**
group_id	Yes	String	Policy group ID.

Parameter	Mandatory	Type	Description
rule	Yes	**AlarmRule** object	Policy trigger rule.

Table 4-86 AlarmRule

Parameter	Mandatory	Type	Description
actions	Yes	Array of **Action** objects	Action executed after a specified policy is matched.
conditions	Yes	Array of **AlarmConditions** objects	Condition contents. A rule can contain multiple conditions in AND relationships. One condition describes the matching method of one metric.

Table 4-87 Action

Parameter	Mandatory	Type	Description
type	Yes	String	scale_out_k8s and scale_out_vm indicate a scale-out. scale_in_k8s and scale_in_vm indicate a scale-in. Note: The value containing k8s indicates a scale-in or -out for containerized applications and that containing vm indicates a scale-in or -out for process applications. Enumeration values: • **scale_out_k8s** • **scale_out_vm** • **scale_in_k8s** • **scale_in_vm**
parameters	Yes	Array of **Parameter** objects	Number of scale-in or -out instances.

Table 4-88 Parameter

Parameter	Mandatory	Type	Description
scale_unit	Yes	Integer	Number of scale-in or -out instances. The value is between the minimum number of instances to the maximum number of instances in a policy group.

Table 4-89 AlarmConditions

Parameter	Mandatory	Type	Description
metric_namespace	Yes	String	Metric namespace. Enumeration values: • **PAAS.CONTAINER** • **PAAS.CUSTOMMETRICS**
metric_name	Yes	String	Metric name. Enumeration values: • **^[a-zA-Z_:][a-zA-Z0-9_:]{0** • **254}$**
metric_unit	Yes	String	Unit. Note: The value is retrieved from an AMS API and varies with the metric name.
period	Yes	Integer	Statistical period (unit: s). Enumeration values: • **20** • **60** • **300** • **900** • **1800** • **3600**
evaluation_periods	Yes	Integer	Number of consecutive periods. Enumeration values: • **1** • **2** • **3** • **4** • **5**

Parameter	Mandatory	Type	Description
statistic	Yes	String	Statistic. Enumeration values: • **average**
metric_operation	Yes	String	Metric operator. Option: > or <. For example, you can use > in a threshold criterion (when the value of a metric is greater than metric_thresholdUpdate) to trigger actions. Enumeration values: • > • <
metric_threshold	Yes	Integer	Threshold condition.

Response Parameters

Status code: 200

Table 4-90 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Details.
policy_id	String	Policy ID.

Example Requests

- Example of an alarm policy

```
POST https://{EndPoint}/v1/{project_id}/pe/policy

{
  "name" : "policy_2",
  "policy_type" : "RECURRENCE",
  "rule" : {
    "conditions" : [ {
      "launch_time" : "13:45",
      "recurrence_type" : "Weekly",
      "recurrence_value" : "0,1,4",
      "start_time" : "2017-01-26T03:33Z",
      "end_time" : "2099-01-31T03:33Z"
    } ],
    "actions" : [ {
      "type" : "scale_set_k8s",
      "parameters" : {
        "scale_unit" : 1
```



```
    }
   } ]
  }
}
```

- Example of a scheduled policy

```
POST https://{EndPoint}/v1/{project_id}/pe/policy

{
  "name" : "policy_1",
  "policy_type" : "ALARM",
  "rule" : {
    "conditions" : [ {
      "metric_namespace" : "PAAS.CONTAINER",
      "metric_name" : "cpuUsage",
      "metric_unit" : "Percent",
      "period" : 60,
      "evaluation_periods" : 1,
      "statistic" : "average",
      "metric_operation" : ">",
      "metric_threshold" : 70
    } ],
    "actions" : [ {
      "type" : "scale_out_k8s",
      "parameters" : {
        "scale_unit" : 1
      }
    } ]
  }
}
```

- Example of a periodic policy

```
POST https://{EndPoint}/v1/{project_id}/pe/policy

{
  "name" : "policy1",
  "policy_type" : "SCHEDULED",
  "rule" : {
    "conditions" : [ {
      "launch_time" : "2017-03-04T03:37Z",
      "recurrence_type" : null,
      "recurrence_value" : null,
      "start_time" : null,
      "end_time" : null
    } ],
    "actions" : [ {
      "type" : "scale_set_k8s",
      "parameters" : {
        "scale_unit" : 1
      }
    } ]
  }
}
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "errorCode" : "SVCSTG.PE.0",
  "errorMessage" : "",
  "policy_id" : "1b9994f0-847a-45e4-aeee-e8b604dddb34"
}
```


Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Invalid request. The client should not repeat the request without modifications.
401	Unauthorized: The authorization information is incorrect or invalid.
403	Forbidden: The request is rejected. The server has received the request and understood it, but the server refuses to respond to it. The client should not repeat the request without modifications.
500	Internal Server Error: The server is able to receive the request but unable to understand the request.
503	Service Unavailable: The requested service is invalid. The client should not repeat the request without modifications.

Error Codes

See **Error Codes**.

4.2.2 Querying the Policy List

Function

This API is used to query details about all policies of a specified project. [It is not supported in versions later than 1.8.0.] (tag:hcs)

URI

GET /v1/{project_id}/pe/policy

Table 4-91 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID

Request Parameters

Table 4-92 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM.
Content-Type	Yes	String	Content type, which is application/json. Enumeration values: • **application/json**
resourcetype	Yes	String	Resource type. Value: nod or app. Enumeration values: • **node** • **app**
Cluster-Id	Yes	String	Cluster ID.

Response Parameters

Status code: 200

Table 4-93 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Details.
policies	Array of **AllPolicyContext** objects	Details about a modified policy.

Table 4-94 AllPolicyContext

Parameter	Type	Description
id	String	Policy ID.
group_id	String	Policy group ID.
name	String	Policy name.
policy_type	String	Policy type.

Parameter	Type	Description
rule	**AllRule** object	Policy trigger rule.
create_time	String	Creation time.
update_time	String	Update time.
status	String	Status.

Table 4-95 AllRule

Parameter	Type	Description
conditions	Array of **AllConditions** objects	Condition contents. When an alarm policy is used, its conditions cannot conflict. Example: You cannot set a metric greater than 10% in one condition and smaller than 20% in another condition.
actions	Array of **Action** objects	Action executed after a specified policy is matched.
name	String	Policy name.

Table 4-96 AllConditions

Parameter	Type	Description
launch_time	String	Time when the policy is triggered. This parameter is available when policy_type is set to SCHEDULED or RECURRENCE.
recurrence_type	String	Period type. This parameter is left blank for scheduled policies. This parameter is available when policy_type is set to SCHEDULED or RECURRENCE.
recurrence_value	String	Specific trigger time of a periodic policy. This parameter is left blank for scheduled policies. This parameter is available when policy_type is set to SCHEDULED or RECURRENCE.
start_time	Integer	Start time of the periodic policy. This parameter is left blank for the scheduled policy. This parameter is available when policy_type is set to SCHEDULED or RECURRENCE.

Parameter	Type	Description
end_time	Integer	End time of the periodic policy. This parameter is left blank for the scheduled policy. This parameter is available when policy_type is set to SCHEDULED or RECURRENCE.
time_zone	String	Time zone. This parameter is available when policy_type is set to SCHEDULED or RECURRENCE.
metric_namespace	String	Metric namespace. This parameter is available when policy_type is set to ALARM. Enumeration values: • **PAAS.CONTAINER** • **PAAS.CUSTOMMETRICS**
metric_name	String	Metric name. This parameter is available when policy_type is set to ALARM. Enumeration values: • **^[a-zA-Z_:][a-zA-Z0-9_:]{0** • **254}$**
metric_unit	String	Unit. Note: The value is retrieved from an AMS API and varies with the metric name. This parameter is available when policy_type is set to ALARM.
period	Integer	Statistical period (unit: s). This parameter is available when policy_type is set to ALARM. Enumeration values: • **20** • **60** • **300** • **900** • **1800** • **3600**
evaluation_periods	Integer	Number of consecutive periods. This parameter is available when policy_type is set to ALARM. Enumeration values: • **1** • **2** • **3** • **4** • **5**

Parameter	Type	Description
statistic	String	Statistic. This parameter is available when policy_type is set to ALARM. Enumeration values: • **average**
metric_operation	String	Metric operator. Option: > or <. For example, you can use > in a threshold criterion (when the value of a metric is greater than metric_thresholdUpdate) to trigger actions. This parameter is available when policy_type is set to ALARM. Enumeration values: • >\n• <
metric_threshold	Integer	Threshold condition. This parameter is available when policy_type is set to ALARM.

Table 4-97 Action

Parameter	Type	Description
type	String	scale_out_k8s and scale_out_vm indicate a scale-out. scale_in_k8s and scale_in_vm indicate a scale-in. Note: The value containing k8s indicates a scale-in or -out for containerized applications and that containing vm indicates a scale-in or -out for process applications. Enumeration values: • **scale_out_k8s**\n• **scale_out_vm**\n• **scale_in_k8s**\n• **scale_in_vm**
parameters	Array of **Parameter** objects	Number of scale-in or -out instances.

Table 4-98 Parameter

Parameter	Type	Description
scale_unit	Integer	Number of scale-in or -out instances. The value is between the minimum number of instances to the maximum number of instances in a policy group.

Example Requests

None

Example Responses

Status code: 200

OK: The request is successful.


```
{
  "errorCode" : "SVCSTG.PE.0",
  "errorMessage" : "",
  "policies" : [ {
    "id" : "8accffb6-e0ed-4433-b216-ccf6960eb1ad",
    "name" : "alarm",
    "group_id" : "77c37e1f-aa0c-438d-8445-39b3997786a2",
    "policy_type" : "ALARM",
    "rule" : {
      "name" : "",
      "conditions" : [ {
        "metric_namespace" : "PAAS.CONTAINER",
        "metric_name" : "cpuCoreLimit",
        "metric_unit" : "Percent",
        "period" : 60,
        "evaluation_periods" : 1,
        "statistic" : "average",
        "metric_operation" : ">",
        "metric_threshold" : 100,
        "metric_dimensions" : null
      } ],
      "actions" : [ {
        "type" : "scale_out_k8s",
        "parameters" : {
          "scale_unit" : 1
        }
      } ]
    },
    "create_time" : "2017-12-21T09:13:42Z",
    "update_time" : "2017-12-21T09:13:42Z",
    "status" : "enabled"
  }, {
    "id" : "9aafbd3d-eac4-4a92-a342-5b6f8d60fff2",
    "name" : "dingshi2",
    "group_id" : "77c37e1f-aa0c-438d-8445-39b3997786a2",
    "policy_type" : "SCHEDULED",
    "rule" : {
      "name" : "",
      "conditions" : [ {
        "launch_time" : "2017-12-22T06:30Z",
        "recurrence_type" : "",
        "recurrence_value" : "",
        "start_time" : "",
        "end_time" : ""
```



```
    } ],
    "actions" : [ {
      "type" : "scale_set_k8s",
      "parameters" : {
        "scale_unit" : 1
      }
    } ]
  },
  "create_time" : "2017-12-21T09:14:00Z",
  "update_time" : "2017-12-21T09:14:00Z",
  "status" : "enabled"
 } ]
}
```


Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Invalid request. The client should not repeat the request without modifications.
401	Unauthorized: The authorization information is incorrect or invalid.
403	Forbidden: The request is rejected. The server has received the request and understood it, but the server refuses to respond to it. The client should not repeat the request without modifications.
500	Internal Server Error: The server is able to receive the request but unable to understand the request.
503	Service Unavailable: The requested service is invalid. The client should not repeat the request without modifications.

Error Codes

See **Error Codes**.

4.2.3 Deleting a Policy

Function

This API is used to delete a policy based on its ID. [It is not supported in versions later than 1.8.0.] (tag:hcs)

URI

DELETE /v1/{project_id}/pe/policy/{policy_id}

Table 4-99 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
policy_id	Yes	String	Policy ID. The policy with this ID is to be deleted.

Request Parameters

Table 4-100 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Project-level token obtained from IAM.
Content-Type	Yes	String	Content type, which is application/json. Enumeration values: • **application/json**

Response Parameters

Status code: 200

Table 4-101 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Details.

Example Requests

Delete the policy whose ID is 1b9994f0-847a-45e4-aeee-e8b604dddb34.

```
DELETE https://{Endpoint}/v1/{project_id}/pe/policy/1b9994f0-847a-45e4-aeee-e8b604dddb34

N/A
```

Example Responses

None

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Invalid request. The client should not repeat the request without modifications.
401	Unauthorized: The authorization information is incorrect or invalid.
403	Forbidden: The request is rejected. The server has received the request and understood it, but the server refuses to respond to it. The client should not repeat the request without modifications.
500	Internal Server Error: The server is able to receive the request but unable to understand the request.
503	Service Unavailable: The requested service is invalid. The client should not repeat the request without modifications.

Error Codes

See **Error Codes**.

4.2.4 Modifying a Policy

Function

This API is used to modify a policy. [It is not supported in versions later than 1.8.0.] (tag:hcs) Alarm policies can be modified, but scheduled and periodic policies cannot.

URI

PUT /v1/{project_id}/pe/policy/{policy_id}

Table 4-102 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
policy_id	Yes	String	Policy ID.

Request Parameters

Table 4-103 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Content type, which is application/json;charset=utf-8. Enumeration values: • **application/json;charset=utf-8** • **application/json**
Cluster-Id	Yes	String	Cluster ID.
Namespace	Yes	String	Namespace.
Deployment-Name	Yes	String	Application name.
X-Auth-Token	Yes	String	Project-level token obtained from IAM.

Table 4-104 Request body parameters

Parameter	Mandatory	Type	Description
id	Yes	String	Policy ID.
name	Yes	String	Policy name, which cannot be modified. The value must contain 1 to 64 characters starting with a letter. Only digits, letters, underscores (_), and hyphens (-) are allowed.
policy_type	Yes	String	Policy type. Currently, only ALARM policies are supported. Enumeration values: • **ALARM**
rule	Yes	**AlarmRule** object	Policy trigger rule.
group_id	Yes	String	Policy group ID.

Table 4-105 AlarmRule

Parameter	Mandatory	Type	Description
actions	Yes	Array of **Action** objects	Action executed after a specified policy is matched.
conditions	Yes	Array of **AlarmConditions** objects	Condition contents. A rule can contain multiple conditions in AND relationships. One condition describes the matching method of one metric.

Table 4-106 Action

Parameter	Mandatory	Type	Description
type	Yes	String	scale_out_k8s and scale_out_vm indicate a scale-out. scale_in_k8s and scale_in_vm indicate a scale-in. Note: The value containing k8s indicates a scale-in or -out for containerized applications and that containing vm indicates a scale-in or -out for process applications. Enumeration values: • **scale_out_k8s** • **scale_out_vm** • **scale_in_k8s** • **scale_in_vm**
parameters	Yes	Array of **Parameter** objects	Number of scale-in or -out instances.

Table 4-107 Parameter

Parameter	Mandatory	Type	Description
scale_unit	Yes	Integer	Number of scale-in or -out instances. The value is between the minimum number of instances to the maximum number of instances in a policy group.

Table 4-108 AlarmConditions

Parameter	Mandatory	Type	Description
metric_namespace	Yes	String	Metric namespace. Enumeration values: • **PAAS.CONTAINER** • **PAAS.CUSTOMMETRICS**
metric_name	Yes	String	Metric name. Enumeration values: • **^[a-zA-Z_:][a-zA-Z0-9_:]{0** • **254}$**
metric_unit	Yes	String	Unit. Note: The value is retrieved from an AMS API and varies with the metric name.
period	Yes	Integer	Statistical period (unit: s). Enumeration values: • **20** • **60** • **300** • **900** • **1800** • **3600**
evaluation_periods	Yes	Integer	Number of consecutive periods. Enumeration values: • **1** • **2** • **3** • **4** • **5**
statistic	Yes	String	Statistic. Enumeration values: • **average**

Parameter	Mandatory	Type	Description
metric_operation	Yes	String	Metric operator. Option: > or <. For example, you can use > in a threshold criterion (when the value of a metric is greater than metric_thresholdUpdate) to trigger actions. Enumeration values: • >\n• <
metric_threshold	Yes	Integer	Threshold condition.

Response Parameters

Status code: 200

Table 4-109 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Details.
context	**PePolicyContext** object	Details about a modified policy.

Table 4-110 PePolicyContext

Parameter	Type	Description
id	String	Policy ID.
name	String	Policy name, which cannot be modified. The value must contain 1 to 64 characters starting with a letter. Only digits, letters, underscores (_), and hyphens (-) are allowed.
policy_type	String	Policy type. Currently, only ALARM policies are supported. Enumeration values: • **ALARM**
rule	**AlarmRule** object	Policy trigger rule.

Table 4-111 AlarmRule

Parameter	Type	Description
actions	Array of **Action** objects	Action executed after a specified policy is matched.
conditions	Array of **AlarmConditions** objects	Condition contents. A rule can contain multiple conditions in AND relationships. One condition describes the matching method of one metric.

Table 4-112 Action

Parameter	Type	Description
type	String	scale_out_k8s and scale_out_vm indicate a scale-out. scale_in_k8s and scale_in_vm indicate a scale-in. Note: The value containing k8s indicates a scale-in or -out for containerized applications and that containing vm indicates a scale-in or -out for process applications. Enumeration values: • **scale_out_k8s** • **scale_out_vm** • **scale_in_k8s** • **scale_in_vm**
parameters	Array of **Parameter** objects	Number of scale-in or -out instances.

Table 4-113 Parameter

Parameter	Type	Description
scale_unit	Integer	Number of scale-in or -out instances. The value is between the minimum number of instances to the maximum number of instances in a policy group.

Table 4-114 AlarmConditions

Parameter	Type	Description
metric_namespace	String	Metric namespace. Enumeration values: • **PAAS.CONTAINER** • **PAAS.CUSTOMMETRICS**
metric_name	String	Metric name. Enumeration values: • **^[a-zA-Z_:][a-zA-Z0-9_:]{0** • **254}$**
metric_unit	String	Unit. Note: The value is retrieved from an AMS API and varies with the metric name.
period	Integer	Statistical period (unit: s). Enumeration values: • **20** • **60** • **300** • **900** • **1800** • **3600**
evaluation_periods	Integer	Number of consecutive periods. Enumeration values: • **1** • **2** • **3** • **4** • **5**
statistic	String	Statistic. Enumeration values: • **average**
metric_operation	String	Metric operator. Option: > or <. For example, you can use > in a threshold criterion (when the value of a metric is greater than metric_thresholdUpdate) to trigger actions. Enumeration values: • **>** • **<**
metric_threshold	Integer	Threshold condition.

Example Requests

Modify an alarm policy.

```
PUT https://{Endpoint}/v1/{project_id}/pe/policy/{policy_id}

{
  "group_id" : "943eba0f-b10a-4066-6261-1857a53500ff",
  "id" : "5c2eecea-32ac-42c0-be30-f73b15d68429",
  "name" : "policy_1",
  "policy_type" : "ALARM",
  "rule" : {
    "conditions" : [ {
      "metric_namespace" : "PAAS.CONTAINER",
      "metric_name" : "cpuUsage",
      "metric_unit" : "Percent",
      "period" : 60,
      "evaluation_periods" : 1,
      "statistic" : "average",
      "metric_operation" : ">",
      "metric_threshold" : 70
    } ],
    "actions" : [ {
      "type" : "scale_out_k8s",
      "parameters" : {
        "scale_unit" : 1
      }
    } ]
  }
}
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "errorCode" : "SVCSTG.PE.0",
  "errorMessage" : "",
  "context" : {
    "id" : "5c2eecea-32ac-42c0-be30-f73b15d68429",
    "name" : "policy_1",
    "policy_type" : "ALARM",
    "rule" : {
      "conditions" : [ {
        "metric_namespace" : "PAAS.CONTAINER",
        "metric_name" : "cpuUsage",
        "metric_unit" : "Percent",
        "period" : 60,
        "evaluation_periods" : 1,
        "statistic" : "average",
        "metric_operation" : ">",
        "metric_threshold" : 70
      } ],
      "actions" : [ {
        "type" : "scale_out_k8s",
        "parameters" : {
          "scale_unit" : 1
        }
      } ]
    }
  }
}
```


Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Invalid request. The client should not repeat the request without modifications.
401	Unauthorized: The authorization information is incorrect or invalid.
403	Forbidden: The request is rejected. The server has received the request and understood it, but the server refuses to respond to it. The client should not repeat the request without modifications.
500	Internal Server Error: The server is able to receive the request but unable to understand the request.
503	Service Unavailable: The requested service is invalid. The client should not repeat the request without modifications.

Error Codes

See **Error Codes**.

4.2.5 Querying a Policy

Function

This API is used to query details about a policy of a specified project. [It is not supported in versions later than 1.8.0.] (tag:hcs)

URI

GET /v1/{project_id}/pe/policy/{policy_id}

Table 4-115 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
policy_id	Yes	String	Policy ID.

Request Parameters

Table 4-116 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token obtained from IAM.
Content-Type	Yes	String	Content type, which is application/json. Enumeration values: • **application/json**

Response Parameters

Status code: 200

Table 4-117 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Details.
policy	**AllPolicyContext** object	Details about a modified policy.

Table 4-118 AllPolicyContext

Parameter	Type	Description
id	String	Policy ID.
group_id	String	Policy group ID.
name	String	Policy name.
policy_type	String	Policy type.
rule	**AllRule** object	Policy trigger rule.
create_time	String	Creation time.
update_time	String	Update time.
status	String	Status.

Table 4-119 AllRule

Parameter	Type	Description
conditions	Array of **AllConditions** objects	Condition contents. When an alarm policy is used, its conditions cannot conflict. Example: You cannot set a metric greater than 10% in one condition and smaller than 20% in another condition.
actions	Array of **Action** objects	Action executed after a specified policy is matched.
name	String	Policy name.

Table 4-120 AllConditions

Parameter	Type	Description
launch_time	String	Time when the policy is triggered. This parameter is available when policy_type is set to SCHEDULED or RECURRENCE.
recurrence_type	String	Period type. This parameter is left blank for scheduled policies. This parameter is available when policy_type is set to SCHEDULED or RECURRENCE.
recurrence_value	String	Specific trigger time of a periodic policy. This parameter is left blank for scheduled policies. This parameter is available when policy_type is set to SCHEDULED or RECURRENCE.
start_time	Integer	Start time of the periodic policy. This parameter is left blank for the scheduled policy. This parameter is available when policy_type is set to SCHEDULED or RECURRENCE.
end_time	Integer	End time of the periodic policy. This parameter is left blank for the scheduled policy. This parameter is available when policy_type is set to SCHEDULED or RECURRENCE.
time_zone	String	Time zone. This parameter is available when policy_type is set to SCHEDULED or RECURRENCE.

Parameter	Type	Description
metric_namespace	String	Metric namespace. This parameter is available when policy_type is set to ALARM. Enumeration values: • **PAAS.CONTAINER** • **PAAS.CUSTOMMETRICS**
metric_name	String	Metric name. This parameter is available when policy_type is set to ALARM. Enumeration values: • **^[a-zA-Z_:][a-zA-Z0-9_:]{0** • **254}$**
metric_unit	String	Unit. Note: The value is retrieved from an AMS API and varies with the metric name. This parameter is available when policy_type is set to ALARM.
period	Integer	Statistical period (unit: s). This parameter is available when policy_type is set to ALARM. Enumeration values: • **20** • **60** • **300** • **900** • **1800** • **3600**
evaluation_periods	Integer	Number of consecutive periods. This parameter is available when policy_type is set to ALARM. Enumeration values: • **1** • **2** • **3** • **4** • **5**
statistic	String	Statistic. This parameter is available when policy_type is set to ALARM. Enumeration values: • **average**

Parameter	Type	Description
metric_operation	String	Metric operator. Option: > or <. For example, you can use > in a threshold criterion (when the value of a metric is greater than metric_thresholdUpdate) to trigger actions. This parameter is available when policy_type is set to ALARM. Enumeration values: • >= • <
metric_threshold	Integer	Threshold condition. This parameter is available when policy_type is set to ALARM.

Table 4-121 Action

Parameter	Type	Description
type	String	scale_out_k8s and scale_out_vm indicate a scale-out. scale_in_k8s and scale_in_vm indicate a scale-in. Note: The value containing k8s indicates a scale-in or -out for containerized applications and that containing vm indicates a scale-in or -out for process applications. Enumeration values: • **scale_out_k8s** • **scale_out_vm** • **scale_in_k8s** • **scale_in_vm**
parameters	Array of **Parameter** objects	Number of scale-in or -out instances.

Table 4-122 Parameter

Parameter	Type	Description
scale_unit	Integer	Number of scale-in or -out instances. The value is between the minimum number of instances to the maximum number of instances in a policy group.

Example Requests

None

Example Responses

Status code: 200

OK: The request is successful.


```
{
  "errorCode" : "SVCSTG.PE.0",
  "errorMessage" : "",
  "policy" : {
    "id" : "8accffb6-e0ed-4433-b216-ccf6960eb1ad",
    "name" : "alarm",
    "group_id" : "77c37e1f-aa0c-438d-8445-39b3997786a2",
    "policy_type" : "ALARM",
    "rule" : {
      "name" : "",
      "conditions" : [ {
        "metric_namespace" : "PAAS.CONTAINER",
        "metric_name" : "cpuCoreLimit",
        "metric_unit" : "Percent",
        "period" : 60,
        "evaluation_periods" : 1,
        "statistic" : "average",
        "metric_operation" : ">",
        "metric_threshold" : 100,
        "metric_dimensions" : null
      } ],
      "actions" : [ {
        "type" : "scale_out_k8s",
        "parameters" : {
          "scale_unit" : 1
        }
      } ]
    },
    "create_time" : "2017-12-21T09:13:42Z",
    "update_time" : "2017-12-21T09:13:42Z",
    "status" : "enabled"
  }
}
```


Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Invalid request. The client should not repeat the request without modifications.
401	Unauthorized: The authorization information is incorrect or invalid.
403	Forbidden: The request is rejected. The server has received the request and understood it, but the server refuses to respond to it. The client should not repeat the request without modifications.
500	Internal Server Error: The server is able to receive the request but unable to understand the request.

Status Code	Description
503	Service Unavailable: The requested service is invalid. The client should not repeat the request without modifications.

Error Codes

See **Error Codes**.

4.2.6 Modifying Policy Group Attributes

Function

This API is used to modify policy group attributes. [It is not supported in versions later than 1.8.0.] (tag:hcs)

URI

PUT /v1/{project_id}/pe/policy/config

Table 4-123 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID

Request Parameters

Table 4-124 Request header parameters

Parameter	Mandatory	Type	Description
ResourceType	Yes	String	Resource type. Default value: app, indicating that the policy group attributes of an application are modified. Enumeration values: • **app**
Cluster-Id	Yes	String	Cluster ID.
Namespace	Yes	String	Namespace.
Deployment-Name	Yes	String	Application name.
X-Auth-Token	Yes	String	Project-level token obtained from IAM.

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Content type, which is application/json. Enumeration values: • **application/json**

Table 4-125 Request body parameters

Parameter	Mandatory	Type	Description
max_instances	No	Integer	Maximum number of instances, indicating the upper limit for capacity expansion.
min_instances	No	Integer	Minimum number of instances, indicating the lower limit for capacity expansion.
cooldown_time	No	Integer	Cooldown period (unit: s). After an applied policy is executed, the next policy can be executed only after the cooldown period expires.
deployment_name	No	String	Application name.

Response Parameters

Status code: 200

Table 4-126 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Details.

Example Requests

Modify policy group attributes.

```
PUT https://{Endpoint}/v1/{project_id}/pe/policy/config

{
  "max_instances" : 100,
  "min_instances" : 1,
  "cooldown_time" : 60,
```



```
  "deployment_name" : "test01"
}
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "errorCode" : "SVCSTG.PE.0",
  "errorMessage" : ""
}
```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Invalid request. The client should not repeat the request without modifications.
401	Unauthorized: The authorization information is incorrect or invalid.
403	Forbidden: The request is rejected. The server has received the request and understood it, but the server refuses to respond to it. The client should not repeat the request without modifications.
500	Internal Server Error: The server is able to receive the request but unable to understand the request.
503	Service Unavailable: The requested service is invalid. The client should not repeat the request without modifications.

Error Codes

See **Error Codes**.

4.2.7 Querying Policy Group Attributes

Function

This API is used to query policy group attributes. [It is not supported in versions later than 1.8.0.] (tag:hcs)

URI

GET /v1/{project_id}/pe/policy/config

Table 4-127 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID

Request Parameters

Table 4-128 Request header parameters

Parameter	Mandatory	Type	Description
ResourceType	Yes	String	Resource type.
Content-Type	Yes	String	Content type, which is application/json;charset=utf-8. Enumeration values: • **application/ json;charset=utf-8** • **application/json**
Cluster-Id	Yes	String	Cluster ID.
Namespace	Yes	String	Namespace.
Deployment-Name	Yes	String	Application name.
X-Auth-Token	Yes	String	Project-level token obtained from IAM.

Response Parameters

Status code: 200

Table 4-129 Response body parameters

Parameter	Type	Description
errorCode	String	Error code.
errorMessage	String	Details.
config	**PolicyConfig** object	Set of policy group attributes.

Table 4-130 PolicyConfig

Parameter	Type	Description
id	String	ID of a policy group attribute.
max_instances	Integer	Maximum number of instances, indicating the upper limit for capacity expansion.
min_instances	Integer	Minimum number of instances, indicating the lower limit for capacity expansion.
cooldown_time	Integer	Cooldown period (unit: s). After an applied policy is executed, the next policy can be executed only after the cooldown period expires.

Example Requests

None

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "errorCode" : "SVCSTG.PE.0",
  "errorMessage" : "",
  "config" : {
    "id" : "f9c7f57e-b1dc-4ef0-a009-ff2848776803",
    "max_instances" : 100,
    "min_instances" : 1,
    "cooldown_time" : 60
  }
}
```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Invalid request. The client should not repeat the request without modifications.
401	Unauthorized: The authorization information is incorrect or invalid.
403	Forbidden: The request is rejected. The server has received the request and understood it, but the server refuses to respond to it. The client should not repeat the request without modifications.
500	Internal Server Error: The server is able to receive the request but unable to understand the request.

Status Code	Description
503	Service Unavailable: The requested service is invalid. The client should not repeat the request without modifications.

Error Codes

See **Error Codes**.

4.3 Log

4.3.1 Querying Logs

Function

This API is used to query logs by different dimensions, such as by cluster, IP address, or application. Pagination queries are supported. For pagination queries, the lineNum (sequence number of the final log in the last query result), type (value: next), and size parameters need to be added. The values of category, searchKey, keyWord, startTime, and endTime must be the same as those in the first query. To implement another pagination query, change the value of lineNum to the sequence number of the final log in the last query result. The rest may be deduced by analogy.

URI

POST /v1/{project_id}/als/action

Table 4-131 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID obtained from IAM. Generally, a project ID contains 32 characters.

Table 4-132 Query Parameters

Parameter	Mandatory	Type	Description
type	Yes	String	API call mode. When the value is querylogs, this API is used to query logs. Enumeration values: • **querylogs**

Request Parameters

Table 4-133 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Project-level token obtained from IAM.
Content-Type	Yes	String	Content type, which is application/json. Enumeration values: • **application/json**

Table 4-134 Request body parameters

Parameter	Mandatory	Type	Description
category	Yes	String	Log type. Options: app_log: application log node_log: host log custom_log: log in a custom path Enumeration values: • **app_log** • **node_log** • **custom_log**
searchKey	Yes	SearchKey object	Log filter criteria, which vary according to log sources.
keyWord	No	String	1. Exact search by keyword is supported. A keyword is a word between two adjacent delimiters. 2. Fuzzy search by keyword is supported. Example: RROR, ERRO?, *ROR*, ERR*, or ER*OR. 3. Exact search by phrase is supported. Example: Start to refresh alm Statistic. 4. Search using AND (&&) or OR (||) is supported. Example: query&&logs or query||logs. Note: Default delimiters include ,'";=()[]{}@&<>/:\n\t\r and spaces.

Parameter	Mandatory	Type	Description
startTime	Yes	Long	Start time of the query (UTC, in ms).
endTime	Yes	Long	End time of the query (UTC, in ms).
lineNum	No	String	Sequence number of the final log in the last query result. This parameter is not required for the first query, but is required for subsequent pagination queries.
type	No	String	Pagination query. This parameter is not required for the first query, but is required for subsequent pagination queries. Enumeration values: • **next**
pageSize/size	No	Integer	Number of logs queried each time. Default value: 5000. Recommended value: 100. For the first query, pageSize is used. For subsequent pagination queries, size is used.
hideSyslog	No	Integer	Whether to hide the system log (icagent\kubectl) during the query. 0 (default): Hide. 1: Display. Enumeration values: • **0** • **1**
isDesc	No	Boolean	Whether to query logs based on lineNum in ascending or descending order. true: lineNum in descending order (from the latest time to the earliest time) false: lineNum in ascending order (from the earliest time to the latest time) Enumeration values: • **true** • **false**

Table 4-135 SearchKey

Parameter	Mandatory	Type	Description
clusterId	Yes	String	CCE cluster: CCE cluster ID Custom cluster: APM Host log: CONFIG_FILE
nameSpace	No	String	CCE cluster namespace.
appName	No	String	Service name.
podName	No	String	Container instance name.
pathFile	No	String	Log file name.
hostIP	No	String	IP address of the VM where logs are located.

Response Parameters

Status code: 200

Table 4-136 Response body parameters

Parameter	Type	Description
errorCode	String	Response code. SVCSTG.ALS.200200: Success response.
errorMessage	String	Response message.
result	**LogsResults** object	Metadata, including results and the total number of returned records.

Table 4-137 LogsResults

Parameter	Type	Description
total	Integer	Number of records that can be returned.
data	Array of **LogItem** objects	Data array.

Table 4-138 LogItem

Parameter	Type	Description
category	String	Log type.
loghash	String	Hash value of the log source.
clusterId	String	CCE cluster ID.
clusterName	String	CCE cluster name.
nameSpace	String	CCE cluster namespace.
podName	String	CCE container instance name.
appName	String	Service name.
serviceID	String	Service ID of an AOM resource.
containerName	String	CCE container name.
logContent	String	Raw log data.
pathFile	String	Absolute path of a log file.
hostIP	String	IP address of the VM where log files are located.
hostId	String	ID of a host in a cluster.
hostName	String	Name of the VM where log files are located.
collectTime	String	Log collection time (UTC time, in ms).
lineNum	String	Sequence number of a log line.
logContentSize	String	Size of a single-line log.

Example Requests

- Query application logs in a cluster.

```
POST https://{Endpoint}/v1/{project_id}/als/action?type=querylogs

{
  "category" : "app_log",
  "searchKey" : {
    "clusterId" : "874819a2-bd6f-11e9-80be-0255ac1001b3"
  },
  "keyWord" : "",
  "startTime" : 1569463658895,
  "endTime" : 1569463958895,
  "pageSize" : 100,
  "hideSyslog" : 0
}
```

- Query data by page.

```
https://{Endpoint}/v1/{project_id}/als/action
```



```
{
  "category" : "app_log",
  "searchKey" : {
    "clusterId" : "874819a2-bd6f-11e9-80be-0255ac1001b3"
  },
  "keyWord" : "",
  "startTime" : 1569463658895,
  "endTime" : 1569463958895,
  "lineNum" : "1569463911294010547",
  "type" : "next",
  "size" : 100,
  "hideSyslog" : 0
}
```

Example Responses

Status code: 200

OK: The request is successful.

```
{
  "errorCode" : "SVCSTR.ALS.200200",
  "errorMessage" : "Query data success",
  "result" : {
    "total" : 5000,
    "data" : [ {
      "category" : "app",
      "loghash" : "496b2070d40a83c17f2625401af8a50aadc316f216771fbe38b94d31feaa30eb",
      "clusterId" : "c693fa7c-54cd-11e8-8055-0255ac101e40",
      "clusterName" : "aomdemo",
      "nameSpace" : "default",
      "podName" : "als0712-7c4875f884-q5wwp",
      "appName" : "als0712",
      "serviceID" : "",
      "containerName" : "container-0",
      "logContent" : "warn:2018/10/09 06:57:01 helloworld.go:108: the main process is running now.n",
      "pathFile" : "/var/paas/sys/log/apm/debug_erro.trace",
      "hostIP" : "192.168.0.133",
      "hostId" : "c11c7211-5a0b-4925-bef4-d078661299b0",
      "hostName" : "192.168.0.133",
      "collectTime" : "1539068233983",
      "lineNum" : "15390682339830002",
      "logContentSize" : "77"
    } ]
  }
}
```

Status Codes

Status Code	Description
200	OK: The request is successful.
400	Bad Request: Invalid request. The client should not repeat the request without modifications.
401	Unauthorized: The authorization information is incorrect or invalid.
403	Forbidden: The request is rejected. The server has received the request and understood it, but the server refuses to respond to it. The client should not repeat the request without modifications.

Status Code	Description
500	Internal Server Error: The server is able to receive the request but unable to understand the request.
503	Service Unavailable: The requested service is invalid. The client should not repeat the request without modifications.

Error Codes

See **Error Codes**.

5 Permissions Policies and Supported Actions

5.1 Introduction

This chapter describes fine-grained permissions management for your AOM. If your cloud account does not need individual Identity and Access Management (IAM) users, then you may skip over this chapter.

By default, new IAM users do not have any permissions assigned. You need to add a user to one or more groups, and assign permissions policies or roles to these groups. The user then inherits permissions from the groups it is a member of. This process is called authorization. After authorization, the user can perform specified operations on AOM.

You can grant users permissions by using roles and policies. Roles are a type of coarse-grained authorization mechanism that defines permissions related to user responsibilities. Policies define API-based permissions for operations on specific resources under certain conditions, allowing for more fine-grained, secure access control of cloud resources.

NOTE

Policy-based authorization is recommended if you want to allow or deny the access to an API.

A cloud account has all of the permissions required to call all APIs, but IAM users must have the required permissions specifically assigned. The permissions required for calling an API are determined by the actions supported by the API. Only users who have been granted permissions can call the API successfully. For example, if an IAM user queries metrics using an API, the user must have been granted permissions that allow the **aom:metric:get** action.

Supported Actions

There are two kinds of policies: system-defined policies and custom policies. If the permissions preset in the system do not meet your requirements, you can create custom policies and apply these policies to user groups for refined access control.

Operations supported by policies are specific to APIs. The following are common concepts related to policies:

- Permissions: Defined by actions in a custom policy.
- APIs: REST APIs that can be called in a custom policy.
- Actions: Added to a custom policy to control permissions for specific operations.
- Dependent actions: Actions on which a specific action depends to take effect. When assigning permissions for the action to a user, you also need to assign permissions for the dependent actions.
- IAM projects and enterprise projects: Type of projects for which an action will take effect. Policies that contain actions supporting both IAM and enterprise projects can be assigned to user groups and take effect in both IAM and Enterprise Management. Policies that only contain actions supporting IAM projects can be assigned to user groups and only take effect for IAM. Such policies will not take effect if they are assigned to user groups in Enterprise Management.

AOM supports the following actions that can be defined in custom policies:

- **Monitoring Actions**: includes the actions supported by monitoring APIs, such as the APIs for querying metrics; querying and adding monitoring data; adding, modifying, querying, and deleting threshold rules; adding, modifying, querying, and deleting application discovery rules.
- **Auto Scaling Actions**: includes the actions supported by auto scaling APIs, such as the APIs for creating, deleting, and modifying policies; querying one or all policies; modifying and querying policy group attributes.
- **Log Actions**: includes the actions supported by log APIs, such as the API for querying logs.

5.2 Monitoring Actions

NOTE

√: supported; x: not supported

Table 5-1 Monitoring actions

Permissions	API	Action	IAM Project	Enterprise Project
Querying metrics	POST /v1/{project_id}/ams/metrics	aom:metric:get	√	×
Querying monitoring data	POST /v1/{project_id}/ams/metricdata	aom:metric:get	√	×
Adding a threshold rule	POST /v1/{project_id}/ams/alarms	aom:alarmRule:create	√	×

Permissions	API	Action	IAM Project	Enterprise Project
Modifying a threshold rule	PUT /v1/{project_id}/ams/alarms	aom:alarmRule:set	√	×
Querying a threshold rule list	GET /v1/{project_id}/ams/alarms	aom:alarmRule:get	√	×
Querying a threshold rule	GET /v1/{project_id}/ams/alarms/{alarm_id}	aom:alarmRule:get	√	×
Deleting a threshold rule	DELETE /v1/{project_id}/ams/alarms/{alarm_id}	aom:alarmRule:delete	√	×
Adding or modifying one or more application discovery rules	PUT /v1/{project_id}/inv/servicediscoveryrules	aom:discoveryRule:set	√	×
Querying an application discovery rule	GET /v1/{project_id}/inv/servicediscoveryrules	aom:discoveryRule:get	√	×
Deleting an application discovery rule	DELETE /v1/{project_id}/inv/servicediscoveryrules	aom:discoveryRule:delete	√	×

5.3 Auto Scaling Actions

NOTE

√: supported; x: not supported

Table 5-2 Auto scaling actions

Permissions	API	Action	IAM Project	Enterprise Project
Creating a policy	POST /v1/{project_id}/pe/policy	aom:autoScalingRule:create	√	×
Deleting a policy	DELETE /v1/{project_id}/pe/policy	aom:autoScalingRule:delete	√	×
Modifying a policy	PUT /v1/{project_id}/pe/policy/{policy_id}	aom:autoScalingRule:update	√	×
Querying a policy list	GET /v1/{project_id}/pe/policy	aom:autoScalingRule:list	√	×
Querying a policy	GET /v1/{project_id}/pe/policy/{policy_id}	aom:autoScalingRule:get	√	×
Modifying policy group attributes	PUT /v1/{project_id}/pe/policy/config	aom:autoScalingRule:update	√	×
Querying policy group attributes	GET /v1/{project_id}/pe/policy/config	aom:autoScalingRule:get	√	×

5.4 Log Actions

NOTE

√: supported; x: not supported

Table 5-3 Log actions

Permissions	API	Action	IAM Project	Enterprise Project
Querying logs	POST /v1/{project_id}/als/action?type=querylogs	aom:log:list	√	×

6 Appendix

6.1 Status Codes

Table 6-1 describes the status codes.

Table 6-1 Status codes

Status Code	Message	Description
100	Continue	The client should continue with its request. This interim response is used to inform the client that the initial part of the requests has been received and not rejected by the server.
101	Switching Protocols	The protocol should be switched. The protocol can only be switched to a more advanced protocol. For example, a new HTTP protocol.
200	OK	The request has succeeded.
201	Created	The request has been fulfilled, resulting in the creation of a new resource.

Status Code	Message	Description
202	Accepted	The request has been accepted, but the processing has not been completed.
203	Non-Authoritative Information	The server successfully processed the request, but is returning information that may be from another source.
204	No Content	The server has successfully processed the request, but does not return any content. The status code is returned in response to an HTTP OPTIONS request.
205	Reset Content	The server has fulfilled the request, but the requester is required to reset the content.
206	Partial Content	The server has successfully processed a part of the GET request.
300	Multiple Choices	There are multiple options for the location of the requested resource. The response contains a list of resource characteristics and addresses from which a user terminal (such as a browser) can choose the most appropriate one.
301	Moved Permanently	The requested resource has been assigned with a new permanent URI. This new URI is contained in the response.
302	Found	The requested resource resides temporarily under a different URI.

Status Code	Message	Description
303	See Other	The response to the request can be found under a different URI, and should be retrieved using a GET or POST method.
304	Not Modified	The requested resource has not been modified. When the server returns this status code, it does not return any resources.
305	Use Proxy	The requested resource must be accessed through a proxy.
306	Unused	This HTTP status code is no longer used.
400	Bad Request	The request is invalid. The client should not repeat the request without modifications.
401	Unauthorized	The authorization information provided by the client is incorrect or invalid.
402	Payment Required	This status code is reserved for future use.
403	Forbidden	The request is rejected. The server has received the request and understood it, but the server is refusing to respond to it. The client should not repeat the request without modifications.
404	Not Found	The requested resource cannot be found. The client should not repeat the request without modifications.

Status Code	Message	Description
405	Method Not Allowed	The method specified in the request is not supported by the requested resource. The client should not repeat the request without modifications.
406	Not Acceptable	The server cannot fulfill the request based on the content characteristics of the request.
407	Proxy Authentication Required	This status code is similar to 401, but indicates that the client must authenticate itself with the proxy first.
408	Request Timeout	The client does not produce a request within the time that the server was prepared to wait. The client may repeat the request without modifications later.
409	Conflict	The request cannot be processed due to a conflict. The resource that the client attempts to create already exits, or the request fails to be processed because of the update of the conflict request.
410	Gone	The requested resource cannot be found. The requested resource has been deleted permanently.
411	Length Required	The server refuses to process the request without a defined Content-Length.

Status Code	Message	Description
412	Precondition Failed	The server does not meet one of the preconditions that the requester puts on the request.
413	Request Entity Too Large	The server refuses to process a request because the request entity is too large. The server may disable the connection to prevent the client from sending requests consecutively. If the server cannot process the request temporarily, the response will contain a Retry-After field.
414	Request-URI Too Long	The request URI is too long for the server to process.
415	Unsupported Media Type	The server cannot process the media format in the request.
416	Requested Range Not Satisfiable	The requested range is invalid.
417	Expectation Failed	The server fails to meet the requirements of the Expect request-header field.
422	Unprocessable Entity	The request is well-formed but is unable to be processed due to semantic errors.

Status Code	Message	Description
429	Too Many Requests	The client sends excessive requests to the server within a given time (exceeding the limit on the access frequency of the client), or the server receives excessive requests within a given time (beyond its processing capability). In this case, the client should repeat requests after the time specified in the Retry-After header of the response expires.
500	Internal Server Error	The server is able to receive the request but unable to understand the request.
501	Not Implemented	The server does not support the function required to fulfill the request.
502	Bad Gateway	The server acting as a gateway or proxy receives an invalid response from a remote server.
503	Service Unavailable	The requested service is invalid. The client should not repeat the request without modifications.
504	Server Timeout	The request cannot be fulfilled within a given time. This status code is returned to the client only when the timeout parameter is specified in the request.
505	HTTP Version Not Supported	The server does not support the HTTP protocol version used in the request.

6.2 Error Codes

If an error occurs in API calling, no result is returned. Identify the causes of errors based on the error codes of each API. If an error occurs in API calling, HTTP status code 4xx or 5xx is returned. The response body contains the specific error code and information. If you are unable to identify the cause of an error, contact technical support and provide the error code to solve the problem.

Format of an Error Response Body

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

```
{
   "errorCode": "SVCSTG_AMS_4000001",
   "errorMessage": "Request param invalid"
}
```

In the response body, **errorCode** is an error code, and **errorMessage** provides information about the error.

Error Code Description

Error Code	Message	Solution
SVCSTG_AMS_4000001	Invalid request parameter.	Check whether the parameter meets requirements.
SVCSTG_AMS_4000002	Invalid namespace.	Check whether the parameter meets requirements.
SVCSTG_AMS_4000003	Dimensions are left blank.	Check whether the parameter meets requirements.
SVCSTG_AMS_4000005	Invalid metric data type.	Check whether the parameter meets requirements.
SVCSTG_AMS_4000006	The metric data value is left blank.	Check whether the parameter meets requirements.
SVCSTG_AMS_4000007	Invalid name or value length in the dimension.	Check whether the parameter meets requirements.
SVCSTG_AMS_4000008	The request exceeds 40 KB.	Check whether the parameter meets requirements.
SVCSTG_AMS_4000009	A metric supports a maximum of 20 dimensions.	Check whether the parameter meets requirements.
SVCSTG_AMS_4000010	Invalid collection time.	Check whether the parameter meets requirements.
SVCSTG_AMS_4000101	Invalid namespace.	Check whether the parameter meets requirements.

Error Code	Message	Solution
SVCSTG_AMS_4000101	Projectid is left blank.	Check whether the parameter meets requirements.
SVCSTG_AMS_4000101	Invalid alarm name.	Check whether the parameter meets requirements.
SVCSTG_AMS_4000102	Invalid inventoryId.	Check whether the parameter meets requirements.
SVCSTG_AMS_4000102	The metric data parameter is null.	Check whether the parameter meets requirements.
SVCSTG_AMS_4000102	The threshold rule name already exists.	Use another name.
SVCSTG_AMS_4000103	ProjectId is left blank.	Check whether the parameter meets requirements.
SVCSTG_AMS_4000103	Invalid period.	Check whether the parameter meets requirements.
SVCSTG_AMS_4000103	Invalid alarm description.	Check whether the parameter meets requirements.
SVCSTG_AMS_4000104	Invalid statistics.	Check whether the parameter meets requirements.
SVCSTG_AMS_4000104	Invalid alarm threshold.	Check whether the parameter meets requirements.
SVCSTG_AMS_4000105	Invalid limit.	Check whether the parameter meets requirements.
SVCSTG_AMS_4000105	Invalid metrics.	Check whether the parameter meets requirements.
SVCSTG_AMS_4000105	Invalid alarm period.	Check whether the parameter meets requirements.
SVCSTG_AMS_4000106	Invalid start.	Check whether the parameter meets requirements.
SVCSTG_AMS_4000106	Invalid time range.	Check whether the parameter meets requirements.
SVCSTG_AMS_4000106	Invalid email list.	Check whether the parameter meets requirements.
SVCSTG_AMS_4000107	The number of data points in a time range exceeds 1440.	Check whether the parameter meets requirements.
SVCSTG_AMS_4000107	The maximum number of threshold rules has been reached.	Contact the administrator.

Error Code	Message	Solution
SVCSTG_AMS_4000108	Invalid time range for alarm queries.	Check whether the parameter meets requirements.
SVCSTG_AMS_4000109	Invalid metricName.	Check whether the parameter meets requirements.
SVCSTG_AMS_4000109	Invalid project ID.	Check whether the parameter meets requirements.
SVCSTG_AMS_4000110	Invalid fillValue.	Check whether the parameter meets requirements.
SVCSTG_AMS_4000110	Invalid limit.	Check whether the parameter meets requirements.
SVCSTG_AMS_4000111	Invalid start.	Check whether the parameter meets requirements.
SVCSTG_AMS_4000115	Invalid request parameter.	Check whether the parameter meets requirements.
SVCSTG_AMS_4000118	Invalid number of consecutive periods.	Check whether the parameter meets requirements.
SVCSTG_AMS_4000119	Invalid alarm statistic.	Check whether the parameter meets requirements.
SVCSTG_AMS_4000120	Invalid alarm comparison operator.	Check whether the parameter meets requirements.
SVCSTG_AMS_4000121	The alarm does not exist.	Check whether the threshold rule exists.
SVCSTG_AMS_5000000	Internal server error.	Contact the administrator.
SVCSTG_AMS_5030001	The Cassandra session is null.	Contact the administrator.
SVCSTG_AMS_5030002	The Cassandra execution is abnormal.	Contact the administrator.
SVCSTG.INV.4000115	Invalid request parameter.	Check the parameter.
SVCSTG.INV.4030000	Forbidden	Use an authorized account.
SVCSTG.INV.5000001	The Elasticsearch session is null.	Contact the administrator.
SVCSTG.INV.5000002	The Elasticsearch execution is abnormal.	Contact the administrator.
SVCSTG.INV.5000003	The call ICMGR is abnormal.	Contact the administrator.

Error Code	Message	Solution
SVCSTG.INV.5000006	The apprule name already exists.	Use another name.
SVCSTG.INV.5000007	The maximum number of rules has been reached.	Delete unnecessary rules and add new rules.
SVCSTG.PE.4001101	Create policy group DeploymentName is invalid.	Check whether the parameter meets requirements.
SVCSTG.PE.40031002	Auth deploymentName failed	Change the workload name to the name of an existing workload for which a scaling rule needs to be created.
SVCSTG.PE.4031012	Failed to verify the project ID.	Check whether the parameter meets requirements.
SVCSTG.PE.4033008	Failed to update the scheduled or periodic policy.	Check whether the parameter meets requirements.
SVCSTG.PE.5001201	Failed to insert or update data in the background.	Contact the administrator.
SVCSTG.PE.5001203	Query error.	Contact the administrator.
SVCSTG.PE.5001205	Failed to delete records.	Contact the administrator.
SVCSTG.PE.5003007	Failed to update the threshold rule.	Contact the administrator.
SVCSTG.PE.4041202	Failed to get record.	Contact the administrator.
SVCSTG.ALS.200.200	Data queried successfully.	-
SVCSTG.ALS.200.201	The maximum length of parameter %s exceeds %s. %s is empty. %s is incorrect.	Check whether the parameter meets requirements.
SVCSTG.ALS.200.203	Failed to query logs.	Check whether the parameter meets requirements.
SVCSTG.ALS.403.105	Invalid project ID.	Check whether the URL project_id and token project_id are the same.
APM.ICMGR.2001401	Privilege Unavailable	Contact the administrator.

6.3 Obtaining an Account ID and Project ID

When making API calls, you may need to enter the username, user ID, project name, and project ID in some URIs. You can obtain them on the **My Credential** page.

Step 1 Log in to the management console.

Step 2 Click the account name in the upper right corner, and choose **My Credential**.

Step 3 On the **My Credentials** page, view the username, user ID, project name, and project ID.

----End