

CodeArts Check

API Reference

Issue 01
Date 2023-11-30



Copyright © Huawei Technologies Co., Ltd. 2023. All rights reserved.

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

Trademarks and Permissions



HUAWEI and other Huawei trademarks are trademarks of Huawei Technologies Co., Ltd.

All other trademarks and trade names mentioned in this document are the property of their respective holders.

Notice

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

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

Huawei Technologies Co., Ltd.

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

Website: <https://www.huawei.com>

Email: support@huawei.com

Security Declaration

Vulnerability

Huawei's regulations on product vulnerability management are subject to "Vul. Response Process". For details about the policy, see the following website:<https://www.huawei.com/en/psirt/vul-response-process>
For enterprise customers who need to obtain vulnerability information, visit:<https://securitybulletin.huawei.com/enterprise/en/security-advisory>

Contents

1 Before You Start.....	1
1.1 Overview.....	1
1.2 Calling Method.....	1
1.3 Endpoints.....	1
1.4 Basic Concepts.....	2
2 API Overview.....	3
3 Calling APIs.....	4
3.1 Making an API Request.....	4
3.2 Authentication.....	7
3.3 Response.....	8
4 APIs.....	10
4.1 Task Management.....	10
4.1.1 Querying the task list.....	10
4.1.2 Stopping a Check Task.....	13
4.1.3 Querying the Task Execution Status.....	15
4.1.4 Querying Historical Scan Results.....	18
4.1.5 Querying Check Parameters of a Task Rule Set (V2).....	22
4.1.6 Task Configuration Check Parameters.....	25
4.1.7 Modifying a Task Rule Set.....	28
4.1.8 Querying Task Check Failure Logs.....	31
4.1.9 Creating a Check Task.....	34
4.1.10 Deleting a Check Task.....	38
4.1.11 Executing a Check Task.....	40
4.1.12 Querying Check Parameters of a Task Rule Set (v3).....	43
4.1.13 Querying the Selected Rule Sets of a Task (V2).....	47
4.1.14 Querying the Selected Rule Sets of a Task (v3).....	49
4.1.15 Obtaining the Directory Tree of a Task.....	52
4.1.16 Shielding a Directory in Task Configuration.....	55
4.1.17 Querying Advanced Settings of a Task.....	58
4.1.18 Configuring Advanced Settings for a Task.....	61
4.2 Defect Management.....	63
4.2.1 Query Defect Summary.....	64

4.2.2 Querying the cmertrics Defect Summary.....	68
4.2.3 Querying Defect Details.....	74
4.2.4 Querying Defect Details Statistics.....	79
4.2.5 Modify Defect Status.....	82
4.3 Rule Management.....	84
4.3.1 Interface for Obtaining the Rule List.....	84
4.3.2 Creating a User-Defined Rule Set.....	88
4.3.3 Querying the Rule Set List.....	92
4.3.4 Viewing the Rule List of a Rule Set.....	96
4.3.5 Deleting a User-Defined Rule Set.....	100
4.3.6 Set the default rule set configuration for the language corresponding to each project.....	102
5 Application Examples.....	105
5.1 Example 1: Querying the Task List.....	105
5.2 Example 2: Querying the Defect Summary.....	107
6 Appendixes.....	110
6.1 Status Codes.....	110
6.2 Error Codes.....	113
6.3 Obtaining a Project ID.....	116
6.4 Obtaining an Account ID.....	116
7 Change History.....	118

1 Before You Start

[1.1 Overview](#)

[1.2 Calling Method](#)

[1.3 Endpoints](#)

[1.4 Basic Concepts](#)

1.1 Overview

CodeArts Check manages code quality in the cloud. It allows you to easily perform static checks and security checks on code in multiple programming languages and obtain comprehensive quality reports. CodeArts Check also allows you to view grouped defects with fix suggestions provided, effectively controlling quality.

This document provides application programming interfaces (APIs) for you to perform CodeArts Check operations, such as creating, deleting, and querying tasks. For details about all supported operations, see [API Overview](#).

Before calling CodeArts Check APIs, ensure that you are familiar with basic concepts ([1.4 Basic Concepts](#)).

1.2 Calling Method

CodeArts Check supports Representational State Transfer (REST) APIs, allowing you to call APIs using HTTPS. For details, see [Calling APIs](#).

1.3 Endpoints

An endpoint is the request address for calling an API. Endpoints vary depending on services and regions.

[Table 1-1](#) lists the CodeArts Check endpoints. Select a desired one based on the service requirements.

Table 1-1 CodeArts Check endpoints

Region	Region ID	Endpoint
EU-Dublin	eu-west-101	check.eu-west-101.myhuaweicloud.eu

1.4 Basic Concepts

- Account
An account is created upon successful registration with Huawei 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 IAM users and grant them permissions for routine management.
- IAM user
An Identity and Access Management (IAM) user is created using an account to use cloud services. Each IAM user has its own identity credentials (password and access keys).
An IAM user can view the account ID and user ID on the My Credentials page of the console. The domain name, username, and password will be required for API authentication.
- Region
Regions are divided from the dimensions of geographical location and network latency. Public services, such as elastic computing, block storage, object storage, VPC network, elastic public IP address, and image, are shared in the same region. Regions are classified into universal and dedicated regions.
 - A universal region provides universal cloud services for common tenants.
 - A dedicated region provides specific services for specific tenants.
- Availability zone (AZ)
An AZ contains one or more physical data centers. Each AZ has independent cooling, fire extinguishing, moisture-proof, and electricity facilities. Within an AZ, compute, network, storage, and other resources are logically divided into multiple clusters. AZs within a region are interconnected using high-speed optical fibers to support cross-AZ high-availability systems.
- Enterprise project
An enterprise project groups and manages enterprise resources in a logically isolated manner. It can contain resources in multiple regions, and allow resources to be added or removed.
For details about how to obtain enterprise project IDs and features, see the [Enterprise Management User Guide](#).

2 API Overview

Table 2-1 CodeArts Check API overview

Type	Subtype	Description
CodeArts Check	Task Management	Task management includes creating, executing, querying, and stopping tasks.
	Defect Management	Defect management includes defect details, such as issue overview, issue status, cyclomatic complexity, and code repetition rate.
	Rule Management	Rule management includes obtaining the rule list APIs, creating user-defined rule sets, and querying the rule set list.

3 Calling APIs

3.1 Making an API Request

3.2 Authentication

3.3 Response

3.1 Making an API Request

This section describes the structure of a REST API request, using 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 requests for calling other APIs.

Request URI

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

A request URI consists of four parts: *{URI-scheme}:// {Endpoint} | {resource-path}? {query-string}*

Parameter	Description
URI-scheme	Protocol used to transmit requests. All APIs use HTTPS .
Endpoint	Domain name or IP address of the server bearing the REST service. The endpoint varies between services in different regions. It can be obtained from 1.3 Endpoints . For example, the endpoint of IAM in the EU-Dublin region is check.eu-west-101.myhuaweicloud.eu .
resource-path	Resource path, that is, the API access path, which is obtained from the URI of a specific API. For example, the resource-path of the API used to obtain a user token is /v3/auth/tokens .

Parameter	Description
query-string	(Optional) Query parameter. The query parameter is prefixed with a question mark (?), in the format of <i>Parameter name=Parameter value</i> . For example, limit=10 indicates that a maximum of 10 data records will be queried.

For example, to obtain the IAM token in the **EU-Dublin** region, obtain the endpoint of this region (check.eu-west-101.myhuaweicloud.eu) and find the **resource-path (/v3/auth/tokens)** in the URI of the API for **obtaining a user token**. Then, construct the URI as follows:

```
https://check.eu-west-101.myhuaweicloud.eu/v3/auth/tokens
```

NOTE

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

Request Methods

HTTP 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, such as 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, when you call the API to obtain a user token, the request method is **POST**, and the request is as follows:

```
POST https://check.eu-west-101.myhuaweicloud.eu/v3/auth/tokens
```

Request Header

You can also add additional header fields to a request, such as fields for specifying a URI and an HTTP method. For example, to request 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 [3.2 Authentication](#).

- **X-Project-ID**: specifies a subproject ID. This parameter is optional. It is used in multi-project scenarios.
- **X-Domain-ID**: identifies an account.

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://check.eu-west-101.myhuaweicloud.eu/v3/auth/tokens
Content-Type: application/json
```

Request Body

A request body is often sent in structured format. It corresponds to **Content-Type** in the request header and passes content except the request header.

The request body varies according to APIs. Certain APIs do not require the request body, such as GET and DELETE.

When you call the API for [obtaining 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 *xxxxxxxxxx* (project ID, for example, eu-west-101) with the actual values. The project ID can be obtained from [1.3 Endpoints](#).

NOTE

The **scope** parameter specifies where a token takes effect. You can set **scope** to an account or a project under an account. In the following example, the token takes effect only for the resources in a specified project. For details about this API, see [Obtaining a User Token Through Password Authentication](#).

```
POST https://check.eu-west-101.myhuaweicloud.eu/v3/auth/tokens
Content-Type: application/json
```

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

```
    }  
  },  
  "scope": {  
    "project": {  
      "name": "xxxxxxx"  
    }  
  }  
}
```

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 requests for calling 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.
- AK/SK-based authentication: Requests are authenticated by encrypting the request body using an AK/SK pair.

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 IAM API used to obtain a user token.

A token is used to acquire temporary permissions. During API authentication using a token, the token is added to requests to get permissions for calling the API.

When calling the API to [obtain a user token](#), you must set **auth.scope** in the request body to **project**.

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

After a token is obtained, add field **X-Auth-Token** to the request header 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:

```
POST https://check.eu-west-101.myhuaweicloud.eu/v3/auth/tokens
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.

AK/SK-based authentication uses AK/SK to sign requests, and the signature is then added to request headers for authentication.

- **AK**: access key ID. It is a unique ID associated with an SK. AK is used together with SK to sign requests.
- **SK**: secret access key. It is used together with an access key ID to identify a sender who initiates a request and to cryptographically sign requests, preventing 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. For details about how to sign requests and use the signing SDK, see [API Request Signing Guide](#).

NOTICE

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 that includes 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 details, see [Status Codes](#).

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

Response Header

Similar to a request, a response also has a header, for example, **content-type**.

Figure 3-1 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 requests for calling other APIs.

Figure 3-1 Header fields of the response to the request for obtaining a user token

```
connection → keep-alive
content-type → application/json
date → Tue, 12 Feb 2019 06:52:13 GMT
server → Web Server
strict-transport-security → max-age=31536000; includeSubdomains;
transfer-encoding → chunked
via → proxy A
x-content-type-options → nosniff
x-download-options → noopen
x-frame-options → SAMEORIGIN
x-iam-trace-id → 218d45ab-d674-4995-af3a-2d0255ba41b5
x-subject-token → [REDACTED] UMC
[REDACTED] 33wEI
[REDACTED] X-
[REDACTED]
x-xss-protection → 1; mode=block;
```

Response Body

A response body is often returned in structured format. It corresponds to **content-type** header field and passes content except the response header.

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

```
{
  "token": {
    "expires_at": "2019-02-13T06:52:13.855000Z",
    "methods": [
      "password"
    ],
    "catalog": [
      {
        "endpoints": [
          {
            "region_id": "eu-west-101",
            .....

```

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

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

In the preceding information, **error_code** is an error code, and **error_msg** describes the error.

4 APIs

- [4.1 Task Management](#)
- [4.2 Defect Management](#)
- [4.3 Rule Management](#)

4.1 Task Management

4.1.1 Querying the task list

Function

This API is used to query the task list of a project based on project_id.

URI

GET /v2/{project_id}/tasks

Table 4-1 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID

Table 4-2 Query Parameters

Parameter	Mandatory	Type	Description
offset	No	Integer	Pagination index, offset Minimum: 0 Default: 0

Parameter	Mandatory	Type	Description
limit	No	Integer	Number of records displayed on each page. A maximum of 100 records can be displayed on each page. Minimum: 1 Maximum: 100 Default: 10

Request Parameters

Table 4-3 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. It can be obtained by calling an IAM API. The value of X-Subject-Token in the response header is the user token. Minimum: 1 Maximum: 100000

Response Parameters

Status code: 200

Table 4-4 Response body parameters

Parameter	Type	Description
tasks	Array of SimpleTaskInfoV2 objects	Task Info
total	Integer	Total

Table 4-5 SimpleTaskInfoV2

Parameter	Type	Description
task_id	String	Specifies the task ID.
task_name	String	Indicates the task name.

Parameter	Type	Description
creator_id	String	Creator ID.
git_url	String	Code Repository Address
git_branch	String	Code repository branch. If the MR mode is used, the value is the source branch.
created_at	String	Created
last_check_time	String	Last Check Time

Status code: 400**Table 4-6** Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Status code: 401**Table 4-7** Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Example Requests

```
GET https://{endpoint}/v2/{task_id}/tasks
```

Example Responses

Status code: 200

Request succeeded!

```
{
  "tasks": [ {
    "task_id": "435b58e*****d01c94a6",
    "task_name": "CSharp_02201119",
    "creator_id": "ecadebb5041***9780f3d905e20",
    "git_url": "git@*****_02201119.git",
    "git_branch": "master",
    "created_at": "2020-02-20 11:21:11",
    "last_check_time": "2020-02-20 11:37:46"
  }
]
```

```
}, {  
  "task_id": "435b58343tg5g36907c1384d01c94a6",  
  "task_name": "CSharp_066",  
  "creator_id": "ecadebb3664*****",  
  "git_url": "git@****_02201119.git",  
  "git_branch": "master",  
  "created_at": "2020-02-20 11:21:11",  
  "last_check_time": "2020-02-20 11:37:46"  
}],  
"total": 2  
}
```

Status code: 400

Bad Request

```
{  
  "error_code": "CC.00000000",  
  "error_msg": "The network is busy. Please try again later."  
}
```

Status code: 401

Unauthorized

```
{  
  "error_code": "CC.00000003",  
  "error_msg": "The authentication information has expired."  
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See [Error Codes](#).

4.1.2 Stopping a Check Task

Function

This interface is used to terminate a check task based on the task ID.

URI

POST /v2/tasks/{task_id}/stop

Table 4-8 Path Parameters

Parameter	Mandatory	Type	Description
task_id	Yes	String	Task ID

Request Parameters

Table 4-9 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	It can be obtained by calling an IAM API. The value of X-Subject-Token in the response header is the user token. Minimum: 1 Maximum: 100000

Response Parameters

Status code: 400

Table 4-10 Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Status code: 401

Table 4-11 Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Example Requests

```
POST https://{endpoint}/v2/tasks/2b31ed520xxxxxebedb6e57xxxxxxx/stop
```

Example Responses

Status code: 200

Request succeeded!

```
null
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "The network is busy. Please try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "The authentication information has expired."
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See [Error Codes](#).

4.1.3 Querying the Task Execution Status

Function

Query the task execution status based on the task ID. Task status: 0 indicates that the task is being checked, 1 indicates that the check fails, 2 indicates that the check is successful, and 3 indicates that the task is stopped. The detailed progress information is displayed only when the check is in progress.

URI

GET /v2/tasks/{task_id}/progress

Table 4-12 Path Parameters

Parameter	Mandatory	Type	Description
task_id	Yes	String	Task ID

Request Parameters

Table 4-13 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	It can be obtained by calling an IAM API. The value of X-Subject-Token in the response header is the user token. Minimum: 1 Maximum: 100000

Response Parameters

Status code: 200

Table 4-14 Response body parameters

Parameter	Type	Description
task_status	Integer	Specifies the task status. The value can be 0 (checking), 1 (check failed), 2 (check succeeded), or 3 (task stopped).
progress	ProgressDetailV2 object	Check Progress

Table 4-15 ProgressDetailV2

Parameter	Type	Description
ratio	String	Progress Percentage
info	String	Progress

Status code: 400

Table 4-16 Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Status code: 401**Table 4-17** Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Example Requests

```
GET https://{endpoint}/v2/tasks/2b31ed520xxxxxxebdb6e57xxxxxxx/progress
```

Example Responses

Status code: 200

Request succeeded!

```
{
  "task_status" : 0,
  "progress" : {
    "ratio" : "50%",
    "info" : "Performing the check..."
  }
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "The network is busy. Please try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "The authentication information has expired."
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See [Error Codes](#).

4.1.4 Querying Historical Scan Results

Function

Provides statistics on the number of issues scanned each time.

URI

GET /v2/{project_id}/tasks/{task_id}/checkrecord

Table 4-18 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
task_id	Yes	String	Task ID

Table 4-19 Query Parameters

Parameter	Mandatory	Type	Description
offset	No	Integer	Pagination index, offset Minimum: 0 Default: 0
limit	No	Integer	Number of records displayed on each page. A maximum of 1000 records can be displayed on each page. Minimum: 1 Maximum: 1000 Default: 10

Parameter	Mandatory	Type	Description
start_time	No	String	Start time for filtering tasks. Tasks are filtered based on the start time.
end_time	No	String	End time of the filtering. The filtering is performed based on the end time of the task check.

Request Parameters

Table 4-20 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. It can be obtained by calling an IAM API. The value of X-Subject-Token in the response header is the user token. Minimum: 1 Maximum: 100000

Response Parameters

Status code: 200

Table 4-21 Response body parameters

Parameter	Type	Description
data	Array of CheckRecordDataInfo objects	Historical data
total	Integer	Total

Table 4-22 CheckRecordDataInfo

Parameter	Type	Description
check_time	String	Schedule

Parameter	Type	Description
check_end_time	String	Checks the task execution end time.
issue_counts	CheckRecordIssueCountsInfo object	Severity

Table 4-23 CheckRecordIssueCountsInfo

Parameter	Type	Description
critical	Integer	Critical Issues
serious	Integer	Major
normal	Integer	Common Problems
prompt	Integer	Suggestion

Status code: 400**Table 4-24** Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Status code: 401**Table 4-25** Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Example Requests

```
GET https://{endpoint}/v2/{project_id}/tasks/{task_id}/checkrecord
```

Example Responses

Status code: 200

Request succeeded!

```
{
  "total" : 100,
  "data" : [ {
    "check_time" : "2020-02-20 11:37:46",
    "check_end_time" : "2020-02-20 11:47:46",
    "issue_counts" : {
      "critical" : 0,
      "serious" : 1,
      "normal" : 2,
      "prompt" : 3
    }
  }, {
    "check_time" : "2020-02-20 11:37:46",
    "check_end_time" : "2020-02-20 11:47:46",
    "issue_counts" : {
      "critical" : 0,
      "serious" : 1,
      "normal" : 2,
      "prompt" : 3
    }
  }
]
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "The network is busy. Please try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "The authentication information has expired."
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See [Error Codes](#).

4.1.5 Querying Check Parameters of a Task Rule Set (V2)

Function

Querying Check Parameters of a Task Rule Set (V2)

URI

GET /v2/{project_id}/tasks/{task_id}/ruleset/{ruleset_id}/check-parameters

Table 4-26 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
task_id	Yes	String	Task ID.
ruleset_id	Yes	String	Rule Set ID

Table 4-27 Query Parameters

Parameter	Mandatory	Type	Description
language	Yes	String	Rule Set Language

Request Parameters

Table 4-28 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	No	String	Setting the Media Type and Encoding Format
X-Auth-Token	Yes	String	User token. It can be obtained by calling an IAM API. The value of X-Subject-Token in the response header is the user token. Minimum: 1 Maximum: 100000

Response Parameters

Status code: 200

Table 4-29 Response body parameters

Parameter	Type	Description
[items]	Array of TaskCheckParameters objects	Request succeeded!

Table 4-30 TaskCheckParameters

Parameter	Type	Description
check_id	Integer	Check Tool ID
name	String	Compilation Parameter Name
checker_configs	Array of CheckConfigInfo objects	Checking Parameter Settings

Table 4-31 CheckConfigInfo

Parameter	Type	Description
name	String	Check the parameter name.
cfg_key	String	Check the key value corresponding to the parameter.
default_value	String	Checking the Default Value of a Parameter
option_value	String	Check the optional parameters.
is_required	Integer	0: optional; 1: mandatory
description	String	Check Parameter Description
type	Integer	Parameter type. The options are as follows: 0: text; 2: optional.
status	String	Parameter status. The options are on (enabled) and off (disabled).
value	String	Checks parameter values.

Status code: 400

Table 4-32 Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Status code: 401**Table 4-33** Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Example Requests

```
GET https://{endpoint}/v2/{project_id}/tasks/{task_id}/ruleset/{ruleset_id}/check-parameters
```

Example Responses

Status code: 200

Request succeeded!

```
[ {  
  "check_id" : 0,  
  "name" : "compile_tool",  
  "check_configs" : [ {  
    "value" : "",  
    "name" : "Command",  
    "cfg_key" : "",  
    "default_value" : "",  
    "option_value" : "",  
    "is_required" : 0,  
    "description" : "Command",  
    "type" : 0,  
    "status" : "on"  
  } ]  
} ]
```

Status code: 400

Bad Request

```
{  
  "error_code" : "CC.00000000",  
  "error_msg" : "The network is busy. Please try again later."  
}
```

Status code: 401

Unauthorized

```
{  
  "error_code" : "CC.00000003",
```

```
"error_msg" : "The authentication information has expired."  
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See [Error Codes](#).

4.1.6 Task Configuration Check Parameters

Function

Task Configuration Check Parameters

URI

POST /v2/{project_id}/tasks/{task_id}/config-parameters

Table 4-34 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
task_id	Yes	String	Task ID

Request Parameters

Table 4-35 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	It can be obtained by calling an IAM API. The value of X-Subject-Token in the response header is the user token. Minimum: 1 Maximum: 100000

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Media type and encoding format.

Table 4-36 Request body parameters

Parameter	Mandatory	Type	Description
check_id	Yes	Integer	Check Tool ID Minimum: 1 Maximum: 11
ruleset_id	Yes	String	Rule Set ID Minimum: 1 Maximum: 64
language	Yes	String	Rule Set Language Minimum: 1 Maximum: 128
status	Yes	String	off: disabled; on: enabled Minimum: 1 Maximum: 3
task_check_settings	Yes	Array of TaskCheckSettingsItem objects	Checking Parameter Information

Table 4-37 TaskCheckSettingsItem

Parameter	Mandatory	Type	Description
cfg_key	Yes	String	Check the key value corresponding to the parameter. Minimum: 1 Maximum: 50
status	Yes	String	Parameter Status Minimum: 1 Maximum: 3
cfg_value	No	String	Checking Parameter Values Minimum: 1 Maximum: 1000

Response Parameters

Status code: 400

Table 4-38 Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Status code: 401

Table 4-39 Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Example Requests

Task configuration check parameter: Check whether the language type of the parameter is csharp and the status is on.

```
POST https://{endpoint}/v2/{project_id}/tasks/{task_id}/config-parameters
```

```
{
  "check_id" : 11,
  "ruleset_id" : "b113bxxxxxxxx5738bee86ecxxxxxxxx",
  "language" : "csharp",
  "status" : "on",
  "task_check_settings" : [ {
    "cfg_key" : "csharpCompileTool",
    "status" : "on",
    "cfg_value" : "msbuild"
  } ]
}
```

Example Responses

Status code: 200

Request succeeded!

```
null
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
```



```
{
  "error_msg": "The network is busy. Please try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code": "CC.00000003",
  "error_msg": "The authentication information has expired."
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See [Error Codes](#).

4.1.7 Modifying a Task Rule Set

Function

Modify a task rule set.

URI

PUT /v2/tasks/{task_id}/ruleset

Table 4-40 Path Parameters

Parameter	Mandatory	Type	Description
task_id	Yes	String	Task ID

Request Parameters

Table 4-41 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Setting the Media Type and Encoding Format

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. It can be obtained by calling an IAM API. The value of X-Subject-Token in the response header is the user token. Minimum: 1 Maximum: 100000

Table 4-42 Request body parameters

Parameter	Mandatory	Type	Description
[items]	Yes	Array of UpdateTaskRulesetItem objects	Modify Rule Set

Table 4-43 UpdateTaskRulesetItem

Parameter	Mandatory	Type	Description
language	Yes	String	Rule Set Language Minimum: 1 Maximum: 128
rule_set_id	Yes	String	Rule Set ID Minimum: 1 Maximum: 50
if_use	Yes	String	Indicates whether the relationship between the task language and rule set is enabled. 1: yes; 0: no. Minimum: 1 Maximum: 1
status	Yes	String	New/Old data. The default value is 1. Minimum: 1 Maximum: 1

Response Parameters

Status code: 400

Table 4-44 Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Status code: 401**Table 4-45** Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Example Requests

Modify the rule set of the task. The language type of the rule set is cpp, the rule setting ID is displayed according to the actual situation, and the rule status is 1.

```
PUT https://{endpoint}/v2/tasks/{task_id}/ruleset
```

```
[ {  
  "language" : "cpp",  
  "rule_set_id" : "7be4cfxxxxxxxxfd8d07077cxxxxxxxx",  
  "if_use" : "1",  
  "status" : "1"  
} ]
```

Example Responses

Status code: 200

Request succeeded!

```
null
```

Status code: 400

Bad Request

```
{  
  "error_code" : "CC.00000000",  
  "error_msg" : "The network is busy. Please try again later."  
}
```

Status code: 401

Unauthorized

```
{  
  "error_code" : "CC.00000003",  
  "error_msg" : "The authentication information has expired."  
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See [Error Codes](#).

4.1.8 Querying Task Check Failure Logs

Function

This interface is used to query task check failure logs. If `execute_id` is not transferred, the latest check log is queried.

URI

GET `/v2/{project_id}/tasks/{task_id}/log-detail`

Table 4-46 Path Parameters

Parameter	Mandatory	Type	Description
<code>project_id</code>	Yes	String	Project ID
<code>task_id</code>	Yes	String	Task ID.

Table 4-47 Query Parameters

Parameter	Mandatory	Type	Description
<code>execute_id</code>	No	String	ID of a single task.

Request Parameters

Table 4-48 Request header parameters

Parameter	Mandatory	Type	Description
<code>Content-Type</code>	No	String	Setting the Media Type and Encoding Format

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. It can be obtained by calling the IAM API (value of X-Subject-Token in the response header). Minimum: 1 Maximum: 100000

Response Parameters

Status code: 200

Table 4-49 Response body parameters

Parameter	Type	Description
param_info	ParamInfo object	Task Details
log_info	Array of LogInfo objects	Log information

Table 4-50 ParamInfo

Parameter	Type	Description
url	String	Repository address
branch	String	Repository Branch
language	String	Repository Language
exclude_dir	String	Excluded Directory
encode	String	Encoding Format
compile_config	String	Compilation Configuration Information
rule_template	String	Rule Set Name

Table 4-51 LogInfo

Parameter	Type	Description
display_name	String	Log Title

Parameter	Type	Description
log	String	Log content
level	String	Log level
analysis	String	Logs
faq	String	Cloud Commerce FAQs

Status code: 400**Table 4-52** Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Status code: 401**Table 4-53** Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Example Requests

```
GET https://{endpoint}/v2/{project_id}/tasks/{task_id}/log-detail
```

Example Responses

Status code: 200

Request succeeded!

```
{
  "param_info": {
    "url": "git@xxxxxx_only00001/file.git",
    "branch": "master",
    "language": "language",
    "exclude_dir": "aa",
    "encode": "",
    "compile_config": "xxx",
    "rule_template": ""
  },
  "log_info": [ {
    "log": "2023-01-04 14:31:33 Running on server:10.75.***.***\n2023-01-04 14:31:34 waiting subJob to execute\n2023-01-04 14:33:38 flush redis cache successfully!\n2023-01-04 14:33:38 all subJob finish!",
    "level": ""
  },
```

```
"analysis" : "2023-01-04 14:31:33 Running on server:10.75.***.***\n2023-01-04 14:31:34 waiting subJob to execute\n2023-01-04 14:33:38 flush redis cache successfully!\n2023-01-04 14:33:38 all subJob finish!",  
"faq" : "",  
"display_name" : ""  
} ]  
}
```

Status code: 400

Bad Request

```
{  
  "error_code" : "CC.00000000",  
  "error_msg" : "The network is busy. Please try again later."  
}
```

Status code: 401

Unauthorized

```
{  
  "error_code" : "CC.00000003",  
  "error_msg" : "The authentication information has expired."  
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See [Error Codes](#).

4.1.9 Creating a Check Task

Function

Create a check task but do not execute it.

URI

POST /v2/{project_id}/task

Table 4-54 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID

Request Parameters

Table 4-55 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the user token. It can be obtained by calling the IAM API (value of X-Subject-Token in the response header). Minimum: 1 Maximum: 100000
Content-Type	Yes	String	Media type and encoding format.

Table 4-56 Request body parameters

Parameter	Mandatory	Type	Description
check_type	No	Array of strings	Check type, in array format. The default value is source.
git_url	Yes	String	Repository address
git_branch	Yes	String	Repository Branch
language	Yes	Array of strings	Check language. The value is in array format and can be cpp,java,js,python,php,css,html,go,typescript,csharp.
rule_sets	No	Array of RuleSetV2 objects	Specifying a Rule Set
task_type	No	String	Check type. The value can be full or inc. full indicates full check, and inc indicates MR check.
username	No	String	Name of the user who has the permission to access the repository.
access_token	No	String	Token of a user who has permission on the repository.
endpoint_id	No	String	Specifies the endpoint ID of the user who has the repository permission.

Parameter	Mandatory	Type	Description
inc_config	No	IncConfigV2 object	Parameters related to incremental check
enable_fossbot	No	Boolean	Indicates whether to enable fossbot check. By default, fossbot check is disabled.
resource_pool_id	No	String	Specifies the resource pool ID, which can be obtained from the resource pool management page.

Table 4-57 RuleSetV2

Parameter	Mandatory	Type	Description
ruleset_id	No	String	Rule set ID, which needs to be obtained from the web page.
language	Yes	String	Check the language. cpp,java,js,python,php,css,html,go,typescript,csharp is supported.

Table 4-58 IncConfigV2

Parameter	Mandatory	Type	Description
parent_task_id	No	String	ID of the parent task to be associated. This parameter is required for pipeline creation or MR creation.
git_source_branch	No	String	Incrementally check code source branches.
git_target_branch	No	String	Incrementally check the code target branch.
merge_id	No	String	Unique MR ID
event_type	No	String	Webhook triggering event type. The options are merge_request and push_request.
action	No	String	Webhook event status, which can be open, close, or update.
title	No	String	MR Title

Response Parameters

Status code: 200

Table 4-59 Response body parameters

Parameter	Type	Description
task_id	String	Specifies the check task ID.

Status code: 400

Table 4-60 Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Status code: 401

Table 4-61 Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Example Requests

Create a check task. The check type of the check task is source, the corresponding code repository URL is displayed based on the site requirements, the code repository branch is master, and the language type is cpp.

```
POST https://{endpoint}/v2/{project_id}/task
{
  "check_type": [ "source" ],
  "git_url": "git@code*****958.git",
  "git_branch": "master",
  "language": [ "cpp", "js" ]
}
```

Example Responses

Status code: 200

Request succeeded!

```
{  
  "task_id" : "d161fd0*****cb3c1d6a783e"  
}
```

Status code: 400

Bad Request

```
{  
  "error_code" : "CC.00000000",  
  "error_msg" : "The network is busy. Please try again later."  
}
```

Status code: 401

Unauthorized

```
{  
  "error_code" : "CC.00000003",  
  "error_msg" : "The authentication information has expired."  
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See [Error Codes](#).

4.1.10 Deleting a Check Task

Function

This interface is used to delete a check task. A task that is being executed cannot be viewed.

URI

DELETE /v2/tasks/{task_id}

Table 4-62 Path Parameters

Parameter	Mandatory	Type	Description
task_id	Yes	String	Check Task ID

Request Parameters

Table 4-63 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the user token. It can be obtained by calling an IAM API. The value of X-Subject-Token in the response header is the user token. Minimum: 1 Maximum: 100000

Response Parameters

Status code: 400

Table 4-64 Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Status code: 401

Table 4-65 Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Example Requests

```
DELETE https://{endpoint}/v2/tasks/{task_id}
```

Example Responses

Status code: 200

Request succeeded!

```
null
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "The network is busy. Please try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "The authentication information has expired."
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See [Error Codes](#).

4.1.11 Executing a Check Task

Function

Execute the check task.

URI

POST /v2/tasks/{task_id}/run

Table 4-66 Path Parameters

Parameter	Mandatory	Type	Description
task_id	Yes	String	Task ID

Request Parameters

Table 4-67 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	It can be obtained by calling an IAM API. The value of X-Subject-Token in the response header is the user token. Minimum: 1 Maximum: 100000
Content-Type	Yes	String	Media type and encoding format.

Table 4-68 Request body parameters

Parameter	Mandatory	Type	Description
username	No	String	Name of the user who has the permission on the temporary repository corresponding to the task.
access_token	No	String	Token of the user who has the permission on the temporary repository corresponding to the task.
git_url	No	String	Temporary repository address corresponding to the task.
git_branch	No	String	Temporary warehouse branch corresponding to the task.

Response Parameters

Status code: 200**Table 4-69** Response body parameters

Parameter	Type	Description
exec_id	String	Execution ID

Status code: 400

Table 4-70 Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Status code: 401**Table 4-71** Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Example Requests

Execute the check task. The URL of the code repository corresponding to the task is displayed based on the site requirements. The branch of the code repository is master, and the user name is displayed based on the site requirements.

```
POST https://{endpoint}/v2/tasks/{taskid}/run
{
  "git_url" : "http://github.xxxxxxx.git",
  "git_branch" : "master",
  "username" : "test",
  "access_token" : "xLD56xxxxxxxxxJF3Sdxxxx"
}
```

Example Responses

Status code: 200

Request succeeded!

```
{
  "exec_id" : "d163535d0*****81d6a7654443e"
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "The network is busy. Please try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
}
```

```
"error_msg" : "The authentication information has expired."  
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See [Error Codes](#).

4.1.12 Querying Check Parameters of a Task Rule Set (v3)

Function

Querying Check Parameters of a Task Rule Set (v3)

URI

GET /v3/{project_id}/tasks/{task_id}/ruleset/{ruleset_id}/check-parameters

Table 4-72 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
task_id	Yes	String	Task ID
ruleset_id	Yes	String	Rule Set ID

Table 4-73 Query Parameters

Parameter	Mandatory	Type	Description
language	Yes	String	Rule Set Language
offset	No	Integer	Pagination index, offset, which is optional. Minimum: 0 Default: 0

Parameter	Mandatory	Type	Description
limit	No	Integer	Number of records displayed on each page. This parameter is optional. Minimum: 1 Maximum: 1000 Default: 10

Request Parameters

Table 4-74 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	No	String	Setting the Media Type and Encoding Format
X-Auth-Token	Yes	String	The user token is obtained by calling the IAM API for obtaining a user token (value of X-Subject-Token in the response header). Minimum: 1 Maximum: 100000

Response Parameters

Status code: 200

Table 4-75 Response body parameters

Parameter	Type	Description
data	Array of TaskCheckParameters objects	Historical data
total	Integer	Total

Table 4-76 TaskCheckParameters

Parameter	Type	Description
check_id	Integer	Check Tool ID

Parameter	Type	Description
name	String	Compilation Parameter Name
checker_configs	Array of CheckConfigInfo objects	Checking Parameter Settings

Table 4-77 CheckConfigInfo

Parameter	Type	Description
name	String	Check the parameter name.
cfg_key	String	Check the key value corresponding to the parameter.
default_value	String	Checking the Default Value of a Parameter
option_value	String	Check the optional parameters.
is_required	Integer	0: optional; 1: mandatory
description	String	Check Parameter Description
type	Integer	Parameter type. The options are as follows: 0: text; 2: optional.
status	String	Parameter status. The options are on (enabled) and off (disabled).
value	String	Checks parameter values.

Status code: 400**Table 4-78** Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Status code: 401**Table 4-79** Response body parameters

Parameter	Type	Description
error_code	String	Error Codes

Parameter	Type	Description
error_msg	String	Description

Example Requests

```
GET https://{endpoint}/v3/{project_id}/tasks/{task_id}/ruleset/{ruleset_id}/check-parameters
```

Example Responses

Status code: 200

Request succeeded!

```
{
  "total": 1,
  "data": [ {
    "check_id": 0,
    "name": "compile_tool",
    "check_configs": [ {
      "value": "",
      "name": "Command",
      "cfg_key": "",
      "default_value": "",
      "option_value": "",
      "is_required": 0,
      "description": "Command",
      "type": 0,
      "status": "on"
    } ]
  } ]
}
```

Status code: 400

Bad Request

```
{
  "error_code": "CC.00000000",
  "error_msg": "The network is busy. Please try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code": "CC.00000003",
  "error_msg": "The authentication information has expired."
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See [Error Codes](#).

4.1.13 Querying the Selected Rule Sets of a Task (V2)

Function

This API is used to query the selected rule set list (v2) of a task.

URI

GET /v2/{project_id}/tasks/{task_id}/rulesets

Table 4-80 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
task_id	Yes	String	Task ID.

Request Parameters

Table 4-81 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	No	String	Setting the Media Type and Encoding Format
X-Auth-Token	Yes	String	User token. It can be obtained by calling an IAM API. The value of X-Subject-Token in the response header is the user token. Minimum: 1 Maximum: 100000

Response Parameters

Status code: 200

Table 4-82 Response body parameters

Parameter	Type	Description
[items]	Array of TaskRulesetInfo objects	Request succeeded!

Table 4-83 TaskRulesetInfo

Parameter	Type	Description
template_id	String	Rule Set ID
language	String	Rule Set Language
template_name	String	Rule Set
type	String	Rule set status. The options are as follows: optional; selected.
status	String	Rule set attribute. The value 0 indicates the default user rule set, and the value 1 indicates the default system rule set.

Status code: 400**Table 4-84** Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Status code: 401**Table 4-85** Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Example Requests

```
GET https://{endpoint}/v2/{project_id}/tasks/{task_id}/rulesets
```

Example Responses

Status code: 200

Request succeeded!

```
[ {
  "template_id" : "396e1c2511744f6fa199d33b26038edd",
  "language" : "Java",
  "template_name" : "Java_cmetrics",
  "type" : "selected",
  "status" : "0"
}, {
  "template_id" : "4e988c481b4c46d0a3297fbe343a662d",
  "language" : "html",
  "template_name" : "General Check Rule Set",
  "type" : "optional",
  "status" : "1"
} ]
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "The network is busy. Please try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "The authentication information has expired."
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See [Error Codes](#).

4.1.14 Querying the Selected Rule Sets of a Task (v3)

Function

This API is used to query the selected rule set list (v3) of a task.

URI

GET /v3/{project_id}/tasks/{task_id}/rulesets

Table 4-86 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
task_id	Yes	String	Task ID

Table 4-87 Query Parameters

Parameter	Mandatory	Type	Description
offset	No	Integer	Pagination index, offset Minimum: 0 Default: 0
limit	No	Integer	Number of records displayed on each page. A maximum of 1000 records can be displayed on each page. Minimum: 1 Maximum: 1000 Default: 10

Request Parameters

Table 4-88 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	No	String	Setting the Media Type and Encoding Format
X-Auth-Token	Yes	String	User token. It can be obtained by calling the IAM API (value of X-Subject-Token in the response header). Minimum: 1 Maximum: 100000

Response Parameters

Status code: 200

Table 4-89 Response body parameters

Parameter	Type	Description
info	Array of TaskRulesetInfo objects	Rule Set Information
total	Integer	Total

Table 4-90 TaskRulesetInfo

Parameter	Type	Description
template_id	String	Rule Set ID
language	String	Rule Set Language
template_name	String	Rule Set
type	String	Rule set status. The options are as follows: optional; selected.
status	String	Rule set attribute. The value 0 indicates the default user rule set, and the value 1 indicates the default system rule set.

Status code: 400**Table 4-91** Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Status code: 401**Table 4-92** Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Example Requests

```
GET https://{endpoint}/v3/{project_id}/tasks/{task_id}/rulesets
```

Example Responses

Status code: 200

Request succeeded!

```
{
  "total" : 1,
  "data" : [ {
    "template_id" : "111111xxxx3246",
    "language" : "java",
    "type" : "selected",
    "template_name" : "General check rule set",
    "status" : "1"
  } ]
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "The network is busy. Please try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "The authentication information has expired."
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See [Error Codes](#).

4.1.15 Obtaining the Directory Tree of a Task

Function

Obtains the directory tree of a task.

URI

GET /v2/{project_id}/tasks/{task_id}/listpathtree

Table 4-93 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
task_id	Yes	String	Task ID.

Table 4-94 Query Parameters

Parameter	Mandatory	Type	Description
current_path	No	String	Directory or file path.
offset	No	Integer	Offset for pagination. Minimum: 0 Default: 0
limit	No	Integer	Number of records displayed on each page (max: 1,000). Minimum: 1 Maximum: 1000 Default: 10

Request Parameters

None

Response Parameters

Status code: 200

Table 4-95 Response body parameters

Parameter	Type	Description
info	Array of TreeNode objects	Directory tree information of a task.
total	Integer	Quantity.

Table 4-96 TreeNode

Parameter	Type	Description
file_name	String	Directory or file name.
file_path	String	Directory or file path.
is_leaf	Boolean	Whether this is a leaf node. The options are true (yes) and false (no).
checkbox_status	String	Shielding status. The value can be unchecked (unshielded), all (all shielded), or half (half shielded).

Status code: 400

Table 4-97 Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Status code: 401

Table 4-98 Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Example Requests

```
GET https://{endpoint}/v2/{project_id}/tasks/{task_id}/listpathtree
```

Example Responses

Status code: 200

Request succeeded!

```
{  
  "info": [{  
    "file_name": ".LAST_RELEASE",  
    "file_path": ".LAST_RELEASE",  
    "is_leaf": true,  
    "checkbox_status": "unchecked"  
  }],  
}
```

```
"total" : 1
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "Network busy. Try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "Network busy. Try again later."
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See [Error Codes](#).

4.1.16 Shielding a Directory in Task Configuration

Function

Shielded directory for task configuration.

URI

POST /v2/{project_id}/tasks/{task_id}/config-ignorepath

Table 4-99 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
task_id	Yes	String	Task ID.

Request Parameters

Table 4-100 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Can be obtained by calling the IAM API for obtaining the user token (the value of X-Subject-Token in the response header). Minimum: 1 Maximum: 100000
Content-Type	Yes	String	Media type and encoding format.

Table 4-101 Request body parameters

Parameter	Mandatory	Type	Description
ignore_path_settings	Yes	Array of IgnorePathSettingItem objects	List of nodes that shield the directory.

Table 4-102 IgnorePathSettingItem

Parameter	Mandatory	Type	Description
file_path	Yes	String	Directory or file path.
checkbox_status	Yes	String	Shielding status. The value can be unchecked (unshielded), all (all shielded), or half (half shielded).

Response Parameters

Status code: 400**Table 4-103** Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Status code: 401**Table 4-104** Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Example Requests

Task configuration shielding directory: The file path is .LAST_RELEASE, and the check status is all.

```
POST https://{endpoint}/v2/{project_id}/tasks/{task_id}/config-ignorepath
{
  "ignore_path_settings" : [ {
    "file_path" : ".LAST_RELEASE",
    "checkbox_status" : "all"
  } ]
}
```

Example Responses**Status code: 200**

Request succeeded!

```
null
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "Network busy. Try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "Authentication information expired."
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request

Status Code	Description
401	Unauthorized

Error Codes

See [Error Codes](#).

4.1.17 Querying Advanced Settings of a Task

Function

Advanced settings of a query task.

URI

GET /v2/{project_id}/tasks/{task_id}/settings

Table 4-105 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
task_id	Yes	String	Task ID.

Table 4-106 Query Parameters

Parameter	Mandatory	Type	Description
offset	No	Integer	(Optional) Offset for pagination. Minimum: 0 Default: 0
limit	No	Integer	(Optional) Number of records displayed on each page. Minimum: 1 Maximum: 1000 Default: 10

Request Parameters

Table 4-107 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Media type and encoding format.
X-Auth-Token	Yes	String	Can be obtained by calling the IAM API for obtaining the user token (the value of X-Subject-Token in the response header). Minimum: 1 Maximum: 100000

Response Parameters

Status code: 200

Table 4-108 Response body parameters

Parameter	Type	Description
info	Array of TaskAdvancedSettings objects	Information about an advanced setting.
total	Integer	Total number.

Table 4-109 TaskAdvancedSettings

Parameter	Type	Description
key	String	Name of an advanced setting.
value	String	Value of an advanced setting.
option_value	String	Options for an advanced setting.
description	String	Description of an advanced setting.

Status code: 400

Table 4-110 Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Status code: 401**Table 4-111** Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Example Requests

```
GET https://{endpoint}/v2/{project_id}/tasks/{task_id}/settings
```

Example Responses

Status code: 200

Request succeeded!

```
{
  "info" : [ {
    "key" : "customImage",
    "value" : "xxxxx",
    "description" : "Custom image.",
    "option_value" : ""
  } ],
  "total" : 1
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "Network busy. Try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "Authentication information expired."
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See [Error Codes](#).

4.1.18 Configuring Advanced Settings for a Task

Function

Configures advanced settings for a task, such as custom image.

URI

POST /v2/{project_id}/tasks/{task_id}/settings

Table 4-112 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID.
task_id	Yes	String	Task ID.

Request Parameters

Table 4-113 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Can be obtained by calling the IAM API for obtaining the user token (the value of X-Subject-Token in the response header). Minimum: 1 Maximum: 100000
Content-Type	Yes	String	Media type and encoding format.

Table 4-114 Request body parameters

Parameter	Mandatory	Type	Description
task_advanced_settings	Yes	Array of TaskAdvancedSettingsItem objects	Information about advanced settings parameters.

Table 4-115 TaskAdvancedSettingsItem

Parameter	Mandatory	Type	Description
key	Yes	String	Name of an advanced setting. Minimum: 1 Maximum: 50
value	Yes	String	Value of an advanced setting. Minimum: 1 Maximum: 1000

Response Parameters

Status code: 400

Table 4-116 Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Status code: 401

Table 4-117 Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Example Requests

Configure advanced options for tasks, such as customizing images.

```
POST https://{endpoint}/v2/{project_id}/tasks/{task_id}/settings

{
  "task_advanced_settings" : [ {
    "key" : "customImage",
    "value" : "xxxx"
  } ]
}
```

Example Responses

Status code: 200

Request succeeded!

```
null
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "Network busy. Try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "Authentication information expired."
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See [Error Codes](#).

4.2 Defect Management

4.2.1 Query Defect Summary

Function

Query the summary of the defect result based on the check task ID. Including the problem overview, problem status, cyclomatic complexity, and code repetition rate.

URI

GET /v2/tasks/{task_id}/defects-summary

Table 4-118 Path Parameters

Parameter	Mandatory	Type	Description
task_id	Yes	String	Task ID

Request Parameters

Table 4-119 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	It can be obtained by calling an IAM API. The value of X-Subject-Token in the response header is the user token. Minimum: 1 Maximum: 100000

Response Parameters

Status code: 200

Table 4-120 Response body parameters

Parameter	Type	Description
task_id	String	Task ID
task_name	String	Indicates the task name.
creator_id	String	Creator ID.
git_url	String	Code Repository Address
git_branch	String	Code repository branch. If the MR mode is used, the value is the source branch.

Parameter	Type	Description
last_check_time	String	Last Check Time
code_line_total	Integer	Total Number of Code Lines
code_line	Integer	Number of Valid Code Lines
code_quality	Number	Code quality
issue_count	Integer	Number of Problems
risk_coefficient	Number	Dangerous Function
duplication_ratio	String	Repetition Ratio
complexity_count	Integer	Complexity
duplicated_lines	Integer	Duplicate Rows
comment_lines	Integer	Number of Comment Lines
comment_ratio	String	Comment Ratio
duplicated_blocks	Integer	Duplicate block
last_exec_time	String	Last Executed
check_type	String	Check type.
created_at	String	Time when the gateway response is created
cyclomatic_complexity_per_method	String	Average Code Complexity
cyclomatic_complexity_per_file	String	Average Code Complexity (File)
critical_count	String	Critical
major_count	String	Major
minor_count	String	Minor
suggestion_count	String	Suggestion

Parameter	Type	Description
is_access	String	Check whether the access control quality is passed.
trigger_type	String	Task Triggering Mode
file_duplication_ratio	String	File repetition rate
new_count	Integer	Number of New Issues
solve_count	Integer	Resolved
duplicated_files	Integer	Number of duplicate files.
new_critical_count	String	Number of New Critical Defects
new_major_count	String	Number of New Major Problems
new_minor_count	String	Number of New Minor Problems
new_suggestion_count	String	Number of New Suggestion Issues

Status code: 400**Table 4-121** Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Status code: 401**Table 4-122** Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Example RequestsGET <https://{endpoint}/v2/tasks/2b31ed520xxxxxxebdb6e57xxxxxxx/defects-summary>

Example Responses

Status code: 200

Request succeeded!

```
{
  "task_id" : "435b58ecf7f54c45907c1384d01c94a6",
  "task_name" : "CSharp_02201119",
  "creator_id" : "ecadebb5041146cc96f9780f3d905e20",
  "git_url" : "git@****_02201119.git",
  "git_branch" : "master",
  "last_check_time" : "2020-02-20 11:37:46",
  "code_line_total" : 20,
  "code_line" : 156,
  "code_quality" : 147.4,
  "issue_count" : 3,
  "risk_coefficient" : 23,
  "duplication_ratio" : "0.0%",
  "complexity_count" : 0,
  "duplicated_lines" : 0,
  "comment_lines" : 0,
  "comment_ratio" : "0",
  "duplicated_blocks" : 0,
  "last_exec_time" : "2 minutes 28 seconds",
  "check_type" : "source",
  "created_at" : "2020-02-20 11:21:11",
  "cyclomatic_complexity_per_method" : "84.0",
  "cyclomatic_complexity_per_file" : "112.0",
  "critical_count" : 2,
  "major_count" : 1,
  "minor_count" : 1,
  "suggestion_count" : 1,
  "is_access" : 0,
  "trigger_type" : 1,
  "file_duplication_ratio" : "33.3%",
  "new_count" : 0,
  "solve_count" : 0,
  "duplicated_files" : 0,
  "new_critical_count" : "0",
  "new_major_count" : "0",
  "new_minor_count" : "0",
  "new_suggestion_count" : "0"
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "The network is busy. Please try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "The authentication information has expired."
}
```


Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See [Error Codes](#).

4.2.2 Querying the cmertrics Defect Summary

Function

Query the cmertrics defect summary based on the check task ID.

URI

GET /v2/{project_id}/tasks/{task_id}/metrics-summary

Table 4-123 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
task_id	Yes	String	Task ID

Request Parameters

Table 4-124 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	The token can be obtained by calling the IAM API used to obtain a user token. The value of X-Subject-Token in the response header is the user token. Minimum: 1 Maximum: 100000

Response Parameters

Status code: 200

Table 4-125 Response body parameters

Parameter	Type	Description
task_id	String	Task ID.
task_name	String	Indicates the task name.
creator_id	String	Creator ID.
git_url	String	Code Repository Address
git_branch	String	Code Repository Branch
last_check_time	String	Last Check Time
last_exec_time	String	Last Executed
check_type	String	Check type.
created_at	String	Created
metric_info	MetricInfo object	Measurement Item

Table 4-126 MetricInfo

Parameter	Type	Description
code_size	String	Code size
raw_lines	String	Total Lines
methods_total	String	Total number of functions
cyclomatic_complexity_total	String	Total Cyclomatic Complexity
cyclomatic_complexity_per_method	String	Average cyclomatic complexity: ≤ 5
maximum_cyclomatic_complexity	String	Maximum Cyclomatic Complexity
huge_cyclomatic_complexity_total	String	Complexity of Ultra-Large Circles

Parameter	Type	Description
huge_cyclomatic_complexity_ratio	String	Ratio of ultra-large cyclomatic complexity
cca_cyclomatic_complexity_total	String	Total number of cca cyclomatic complexity
cca_cyclomatic_complexity_per_method	String	cca average cyclomatic complexity
maximum_cca_cyclomatic_complexity	String	Maximum Cyclomatic Complexity of cca
huge_cca_cyclomatic_complexity_total	String	Total Number of Functions with Ultra-Large Cyclomatic Complexity
cyclomatic_complexity_adequacy	String	Cyclomatic Complexity Satisfaction
maximum_depth	String	Maximum depth
huge_depth_total	String	Number of ultra-large depths
huge_depth_ratio	String	Percentage of Ultra-Large Depth
method_lines	String	Total number of rows in a function
lines_per_method	String	Average Code Lines of a Function
huge_method_total	String	Number of oversized functions
huge_method_ratio	String	Percentage of ultra-large functions
files_total	String	Total Files
folders_total	String	Total number of directories
lines_per_file	String	Average Lines of Code in a File
huge_headerfile_total	String	Number of oversized header files
huge_headerfile_ratio	String	Percentage of Oversized Header Files

Parameter	Type	Description
huge_non_headerfile_total	String	Number of oversized source files
huge_non_headerfile_ratio	String	Percentage of Oversized Source Files
huge_folder_total	String	Number of ultra-large directories
huge_folder_ratio	String	Percentage of Ultra-Large Directories
file_duplication_total	String	Number of duplicate files
file_duplication_ratio	String	File repetition rate
non_hfile_duplication_total	String	Number of duplicate source files
non_hfile_duplication_ratio	String	Source File Repetition Rate
code_duplication_total	String	Number of repeated codes
code_duplication_ratio	String	Duplication Rate
non_hfile_code_duplication_total	String	Number of duplicate source file codes.
non_hfile_code_duplication_ratio	String	Source File Code Repetition Rate
unsafe_functions_total	String	Total Number of Dangerous Functions
unsafe_functions_kloc	String	Density of Dangerous Functions
redundant_code_total	String	Number of redundant codes
redundant_code_kloc	String	Density of redundant code blocks
warning_suppression_total	String	Number of suppressed alarms
warning_suppression_kloc	String	Suppressed Alarm Density

Status code: 400**Table 4-127** Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Status code: 401**Table 4-128** Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Example Requests

```
GET https://{endpoint}/v2/{project_id}/tasks/{task_id}/metrics-summary
```

Example Responses**Status code: 200**

Request succeeded!

```
{
  "task_id" : "435b58ecf7f54c45907c1384d01c94a6",
  "task_name" : "CSharp_02201119",
  "creator_id" : "ecadebb5041146cc96f9780f3d905e20",
  "git_url" : "git@****_02201119.git",
  "git_branch" : "master",
  "last_check_time" : "2020-02-20 11:37:46",
  "last_exec_time" : "2020-02-20 11:37:46",
  "check_type" : "source",
  "created_at" : "2020-02-20 11:21:11",
  "metric_info" : {
    "code_size" : "133426.00",
    "raw_lines" : "182082.00",
    "methods_total" : "6840.00",
    "cyclomatic_complexity_total" : "27435.00",
    "cyclomatic_complexity_per_method" : "7.43",
    "maximum_cyclomatic_complexity" : "346.00",
    "huge_cyclomatic_complexity_total" : "167.00",
    "huge_cyclomatic_complexity_ratio" : "4.13",
    "cca_cyclomatic_complexity_total" : "26693.00",
    "cca_cyclomatic_complexity_per_method" : "7.25",
    "maximum_cca_cyclomatic_complexity" : "342.00",
    "huge_cca_cyclomatic_complexity_total" : "148.00",
    "cyclomatic_complexity_adequacy" : "21.00",
    "maximum_depth" : "333.00",
```

```
"huge_depth_total" : "8.85",
"huge_depth_ratio" : "91546.00",
"method_lines" : "24.91",
"lines_per_method" : "8.04",
"huge_method_total" : "12300",
"huge_method_ratio" : "12300",
"files_total" : "629.00",
"folders_total" : "426.00",
"lines_per_file" : "12300",
"huge_headerfile_total" : "12300",
"huge_headerfile_ratio" : "12300",
"huge_non_headerfile_total" : "12300",
"huge_non_headerfile_ratio" : "12300",
"huge_folder_total" : "12300",
"huge_folder_ratio" : "12300",
"file_duplication_total" : "12300",
"file_duplication_ratio" : "12300",
"non_hfile_duplication_total" : "12300",
"non_hfile_duplication_ratio" : "12300",
"code_duplication_total" : "3.28",
"code_duplication_ratio" : "11782.00",
"non_hfile_code_duplication_total" : "8.59",
"non_hfile_code_duplication_ratio" : "11782.00",
"unsafe_functions_total" : "17.17",
"unsafe_functions_kloc" : "0.00",
"redundant_code_total" : "0.00",
"redundant_code_kloc" : "674.00",
"warning_suppression_total" : "17.00",
"warning_suppression_kloc" : "0.45"
}
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "The network is busy. Please try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "The authentication information has expired."
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See [Error Codes](#).

4.2.3 Querying Defect Details

Function

This API is used to query defect result details by check task ID in pagination mode.

URI

GET /v2/tasks/{task_id}/defects-detail

Table 4-129 Path Parameters

Parameter	Mandatory	Type	Description
task_id	Yes	String	Task ID.

Table 4-130 Query Parameters

Parameter	Mandatory	Type	Description
offset	No	Integer	Pagination index, offset Minimum: 0 Default: 0
limit	No	Integer	Number of records displayed on each page. A maximum of 100 records can be displayed on each page. Minimum: 1 Maximum: 100 Default: 10
status_ids	No	String	Filter by Issue Status
severity	No	String	Indicates the severity. 0: critical; 1: major; 2: minor; 3: warning.

Request Parameters

Table 4-131 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	It can be obtained by calling an IAM API. The value of X-Subject-Token in the response header is the user token. Minimum: 1 Maximum: 100000

Response Parameters

Status code: 200**Table 4-132** Response body parameters

Parameter	Type	Description
defects	Array of DefectInfoV2 objects	Defect Details
total	Integer	Total

Table 4-133 DefectInfoV2

Parameter	Type	Description
defect_id	String	ID of a defect.
defect_checker_name	String	Name of the check item corresponding to the defect
defect_status	String	Defect status. The options are as follows: 0: resolved; 1: resolved; 2: ignored.
rule_system_tags	String	Rule tag. Multiple tags are separated by commas (,).
rule_id	String	Rule ID.
rule_name	String	NATRuleName
line_number	String	Line No. of the File Where the Defect Is Located
defect_content	String	Defect Description

Parameter	Type	Description
defect_level	String	Defect level. The options are as follows: 0: critical; 1: major; 2: minor; 3: warning.
file_path	String	Defect File Path
created_at	String	Creation Time
issue_key	String	Unique ID of a question.
fragment	Array of DefectFragmentV2 objects	Defect Code Snippet Details
events	Array of DefectEvents objects	Call chain information

Table 4-134 DefectFragmentV2

Parameter	Type	Description
line_num	String	Line number
line_content	String	Content of the Code Line
start_offset	Integer	Defect Start Column No.
end_offset	Integer	Defect End Column No.

Table 4-135 DefectEvents

Parameter	Type	Description
events	Array of DefectEvents objects	Call chain information
description	String	Description
fix_suggestions	Array of strings	Solution
line	Integer	Line that has a defect
end_line	Integer	Line where a defect ends
code_context_start_line	Integer	Line where a defect starts
main	Boolean	Code snippet
path	String	Path

Parameter	Type	Description
tag	String	Tag
main_buggy_code	String	Code snippet corresponding to the main event
code_context	String	Code Context

Status code: 400**Table 4-136** Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Status code: 401**Table 4-137** Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Example Requests

```
GET https://{endpoint}/v2/tasks/2b31ed520xxxxxbedb6e57xxxxxxx/defects-summary
```

Example Responses

Status code: 200

Request succeeded!

```
{
  "defects": [ {
    "line_number": "81",
    "defect_id": "5b71a4594bcb4a69a33f576c3564e6cc",
    "rule_system_tags": "fossscan",
    "rule_id": "111",
    "rule_name": "FossScan.1 OpenSource Software",
    "defect_checker_name": "Disallow fallthrough of `case` statements",
    "defect_status": "0",
    "defect_content": "Update or refactor this function so that its implementation doesn't duplicate the one on line 60.",
    "defect_level": "1",
    "file_path": "code/types/functions.ts",
    "created_at": "2020-07-21T02:12:01Z",
    "issue_key": "AA76E96XXXXXXXXXXXXXXXXXXCAAE5",
  }
]
```

```
"fragment" : [ {
  "line_num" : "78",
  "line_content" : " export function padding(all: number);",
  "start_offset" : -1,
  "end_offset" : -1
}, {
  "line_num" : "79",
  "line_content" : " export function padding(topAndBottom: number, leftAndRight: number);",
  "start_offset" : -1,
  "end_offset" : -1
}, {
  "line_num" : "80",
  "line_content" : " export function padding(top: number, right: number, bottom: number, left:
number);",
  "start_offset" : -1,
  "end_offset" : -1
}, {
  "line_num" : "81",
  "line_content" : " export function padding(a: number, b?: number, c?: number, d?: number) {",
  "start_offset" : 11,
  "end_offset" : 19
}, {
  "line_num" : "82",
  "line_content" : " if (b === undefined && c === undefined && d === undefined) {",
  "start_offset" : -1,
  "end_offset" : -1
} ]
}, {
  "line_number" : "190",
  "defect_id" : "5b71a4594wtrfsgrew546t466c3564e6cc",
  "rule_system_tags" : "fossscan",
  "rule_id" : "112",
  "rule_name" : "FossScan.1 OpenSource Software",
  "defect_checker_name" : "Disallow fallthrough of `case` statements",
  "defect_status" : "0",
  "defect_content" : "Update or refactor this function so that its implementation doesn't duplicate the one
on line 173.",
  "defect_level" : "1",
  "file_path" : "code/types/types.ts",
  "created_at" : "2020-07-21T02:12:01Z",
  "issue_key" : "AA76E96XXXXXXXXXXXXXXXXXXXXFE3B6",
  "fragment" : [ {
    "line_num" : "187",
    "line_content" : " var _value;",
    "start_offset" : -1,
    "end_offset" : -1
  }, {
    "line_num" : "188",
    "line_content" : " function getOrSet(): number;",
    "start_offset" : -1,
    "end_offset" : -1
  }, {
    "line_num" : "189",
    "line_content" : " function getOrSet(value: number);",
    "start_offset" : -1,
    "end_offset" : -1
  }, {
    "line_num" : "190",
    "line_content" : " function getOrSet(value?: number) {\\"",
    "start_offset" : 4,
    "end_offset" : 12
  }, {
    "line_num" : "191",
    "line_content" : " if (value === undefined) {\\"",
    "start_offset" : -1,
    "end_offset" : -1
  } ]
} ],
```

```
"total" : 2  
}
```

Status code: 400

Bad Request

```
{  
  "error_code" : "CC.00000000",  
  "error_msg" : "The network is busy. Please try again later."  
}
```

Status code: 401

Unauthorized

```
{  
  "error_code" : "CC.00000003",  
  "error_msg" : "The authentication information has expired."  
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See [Error Codes](#).

4.2.4 Querying Defect Details Statistics

Function

This interface is used to query defect details based on the check task ID.

URI

GET /v2/tasks/{task_id}/defects-statistic

Table 4-138 Path Parameters

Parameter	Mandatory	Type	Description
task_id	Yes	String	Task ID

Request Parameters

Table 4-139 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	It can be obtained by calling an IAM API. The value of X-Subject-Token in the response header is the user token. Minimum: 1 Maximum: 100000

Response Parameters

Status code: 200

Table 4-140 Response body parameters

Parameter	Type	Description
severity	StatisticSeverityV2 object	Problem Level Statistics
status	StatisticStatusV2 object	Issue Status Statistics

Table 4-141 StatisticSeverityV2

Parameter	Type	Description
critical	Integer	Critical
major	Integer	Major
minor	Integer	Minor
suggestion	Integer	Suggestion

Table 4-142 StatisticStatusV2

Parameter	Type	Description
unresolved	Integer	Unsolved
resolved	Integer	Solved
dismissed	Integer	Ignored

Status code: 400**Table 4-143** Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Status code: 401**Table 4-144** Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Example Requests

```
GET https://{endpoint}/v2/tasks/2b31ed520xxxxxxebdb6e57xxxxxxx/defects-statistic
```

Example Responses

Status code: 200

Request succeeded!

```
{
  "severity" : {
    "critical" : 120,
    "major" : 877,
    "minor" : 79,
    "suggestion" : 3
  },
  "status" : {
    "unresolved" : 877,
    "resolved" : 79,
    "dismissed" : 5
  }
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "The network is busy. Please try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
```

```
"error_msg" : "The authentication information has expired."  
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See [Error Codes](#).

4.2.5 Modify Defect Status

Function

Change the status of detected defects to Resolved or Ignored.

URI

PUT /v2/tasks/{task_id}/defect-status

Table 4-145 Path Parameters

Parameter	Mandatory	Type	Description
task_id	Yes	String	Task ID

Request Parameters

Table 4-146 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	It can be obtained by calling an IAM API. The value of X-Subject-Token in the response header is the user token. Minimum: 1 Maximum: 100000
Content-Type	Yes	String	Media type and encoding format.

Table 4-147 Request body parameters

Parameter	Mandatory	Type	Description
defect_id	Yes	String	Question ID. Use commas (,) to separate multiple question IDs.
defect_status	Yes	String	Status. The options are as follows: 2: ignored; 1: resolved; 0: not resolved.

Response Parameters

Status code: 400

Table 4-148 Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Status code: 401

Table 4-149 Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Example Requests

Change the defect status. The defect status ID is displayed based on the site requirements, and the defect status is changed to 1.

```
PUT https://{endpoint}/v2/tasks/{taskid}/defect-status
{
  "defect_id" : "363540xxxxxxxx5105099944xxxxxxxx",
  "defect_status" : "1"
}
```

Example Responses

Status code: 200

Request succeeded!


```
null
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "The network is busy. Please try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "The authentication information has expired."
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See [Error Codes](#).

4.3 Rule Management

4.3.1 Interface for Obtaining the Rule List

Function

This API queries the rule list based on conditions such as the language and problem level.

URI

GET /v2/rules

Table 4-150 Query Parameters

Parameter	Mandatory	Type	Description
rule_languages	No	String	Language corresponding to the rule.
rule_severity	No	String	Defect level. The options are as follows: 0: critical; 1: major; 2: minor; 3: warning.
offset	No	Integer	Pagination index, offset Minimum: 0 Default: 0
limit	No	Integer	Number of records displayed on each page. Minimum: 1 Maximum: 1000 Default: 100

Request Parameters

Table 4-151 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. It can be obtained by calling an IAM API. The value of X-Subject-Token in the response header is the user token. Minimum: 1 Maximum: 100000

Response Parameters

Status code: 200

Table 4-152 Response body parameters

Parameter	Type	Description
info	Array of RuleListItem objects	Parameter description
total	Integer	Total

Table 4-153 RuleListItem

Parameter	Type	Description
rule_set	String	Rule Set Specification Classification
rule_id	String	Rule ID.
rule_language	String	Rule Language
rule_name	String	Rule Name
rule_severity	String	Rule Problem Severity
rule_tags	String	Rule Tag
right_example	String	Correct Example
error_example	String	Negative example
revise_opinion	String	Resolution suggestions
rule_desc	String	Description

Status code: 400**Table 4-154** Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Status code: 401**Table 4-155** Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Example Requests

```
GET https://{endpoint}/v2/rules
```

Example Responses

Status code: 200

Request succeeded!

```
{
  "info" : [ {
    "rule_id" : "2561",
    "rule_language" : "Java",
    "rule_name" : "\"\"=\"+\" cannot be replaced by \"\"=\"+\".",
    "rule_severity" : "1",
    "rule_tags" : "bug",
    "rule_desc" : "<p> uses the constant operator (<code>===</code>) to compare different types. The result
is always <code>>false</code>. </p>",
    "right_example" : "<pre>\nvar a = 8;\nvar b = \"8\";\n\nif (a == b) {\n // ...\n}\n</pre>\n<p>or</p>
\n<pre>\nvar a = 8;\nvar b = \"8\";\n\nif (a === Number(b)) {\n // ...\n}\n</pre>",
    "error_example" : "<pre>\nvar a = 8;\nvar b = \"8\";\n\nif (a === b) { //The comparison result is always
false.\n // ...\n}\n</pre>",
    "revise_opinion" : "None",
    "rule_set" : "HuaWeiJava"
  }, {
    "rule_id" : "8048",
    "rule_language" : "Java",
    "rule_name" : "\"Arrays.stream\" should be used for primitive arrays",
    "rule_severity" : "1",
    "rule_tags" : "performance",
    "rule_desc" : "<p> uses the constant operator (<code>===</code>) to compare different types. The result
is always <code>>false</code>. </p>",
    "right_example" : "<pre>\nvar a = 8;\nvar b = \"8\";\n\nif (a == b) {\n // ...\n}\n</pre>\n<p>or</p>
\n<pre>\nvar a = 8;\nvar b = \"8\";\n\nif (a === Number(b)) {\n // ...\n}\n</pre>",
    "error_example" : "<pre>\nvar a = 8;\nvar b = \"8\";\n\nif (a === b) { //The comparison result is always
false.\n // ...\n}\n</pre>",
    "revise_opinion" : "None",
    "rule_set" : "HuaWeiJava"
  } ],
  "total" : 2
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "The network is busy. Please try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "The authentication information has expired."
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See [Error Codes](#).

4.3.2 Creating a User-Defined Rule Set

Function

Rules can be flexibly combined as required.

URI

POST /v2/ruleset

Request Parameters

Table 4-156 Request header parameters

Parameter	Mandatory	Type	Description
Content-Type	Yes	String	Setting the Media Type and Encoding Format
X-Auth-Token	Yes	String	User token. It can be obtained by calling an IAM API. The value of X-Subject-Token in the response header is the user token. Minimum: 1 Maximum: 100000

Table 4-157 Request body parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID Minimum: 1 Maximum: 50
template_name	Yes	String	New Rule Set Name Minimum: 1 Maximum: 500
language	Yes	String	Rule Set Language Minimum: 1 Maximum: 128

Parameter	Mandatory	Type	Description
is_default	Yes	String	If there is a rule set based on, the value is 1. If there is no rule set based on, the value is 0. Minimum: 1 Maximum: 1
rule_ids	Yes	String	ID of the newly enabled rule. Minimum: 1 Maximum: 4000
uncheck_ids	No	String	ID of the new closure rule. Minimum: 1 Maximum: 4000
template_id	No	String	Rule Set ID Minimum: 1 Maximum: 50
custom_attributes	No	Array of CustomAttributes objects	Users can customize rule parameters and modify rule thresholds.

Table 4-158 CustomAttributes

Parameter	Mandatory	Type	Description
attribute	No	String	Indicates the configuration item attribute. severity indicates the problem severity.
rules	No	Array of CustomAttributesRule objects	Rule Details

Table 4-159 CustomAttributesRule

Parameter	Mandatory	Type	Description
rule_id	No	String	Rule ID.
value	No	String	Attribute problem level. The options are as follows: 0: critical; 1: major; 2: minor; 3: warning.

Parameter	Mandatory	Type	Description
rule_config_list	No	Array of RuleConfig objects	Rule Threshold Details

Table 4-160 RuleConfig

Parameter	Mandatory	Type	Description
id	No	Integer	Rule Configuration ID
rule_id	No	Integer	Rule ID.
default_value	No	String	Default Value
option_value	No	String	Before acceleration
option_key	No	String	Key of the current rule configuration item.
option_name	No	String	Name of the current rule configuration item
template_id	No	String	Rule Set ID
description	No	String	Description

Response Parameters

Status code: 201

Table 4-161 Response body parameters

Parameter	Type	Description
template_id	String	Rule Set ID

Status code: 400

Table 4-162 Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Status code: 401

Table 4-163 Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Example Requests

Create a user-defined rule set. The project ID and template ID corresponding to the rule set are displayed based on the site requirements. The template name is test_cpp, and the language is cpp.

```
POST https://{endpoint}/v2/ruleset
{
  "project_id" : "24b97exxxxxxxxxfb912625b14cxxxxxx",
  "template_id" : "da303dxxxxxxxxaca60dbcc2e2xxxxxx",
  "template_name" : "test_cpp",
  "language" : "cpp",
  "is_default" : "1",
  "rule_ids" : "8139,8138",
  "uncheck_ids" : "1101,1102"
}
```

Example Responses

Status code: 201

Request succeeded!

```
{
  "template_id" : "c53417a4804f45eba9c11991131c9e79"
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "The network is busy. Please try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "The authentication information has expired."
}
```

Status Codes

Status Code	Description
201	Request succeeded!

Status Code	Description
400	Bad Request
401	Unauthorized

Error Codes

See [Error Codes](#).

4.3.3 Querying the Rule Set List

Function

This API is used to query the rule set list based on conditions such as the project ID and language.

URI

GET /v2/{project_id}/rulesets

Table 4-164 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID

Table 4-165 Query Parameters

Parameter	Mandatory	Type	Description
category	No	String	Rule Set Type '0' indicates that only the system rule set is queried. 1 indicates that only the user-defined rule set is queried. 2 indicates that only the user-defined rule sets are queried. '0,1,2' or 'Query all'
offset	No	Integer	Pagination index, offset Minimum: 0 Default: 0

Parameter	Mandatory	Type	Description
limit	No	Integer	Number of records displayed on each page. Minimum: 1 Maximum: 1000 Default: 100

Request Parameters

Table 4-166 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. It can be obtained by calling an IAM API. The value of X-Subject-Token in the response header is the user token. Minimum: 1 Maximum: 100000

Response Parameters

Status code: 200

Table 4-167 Response body parameters

Parameter	Type	Description
info	Array of RulesetItem objects	Rule Set List Information
total	Integer	Total

Table 4-168 RulesetItem

Parameter	Type	Description
template_id	String	Rule Set ID
language	String	Rule Set Language
template_name	String	Rule Set

Parameter	Type	Description
creator_id	String	Creator ID.
creator_name	String	Creator name
template_create_time	String	Creator Time
is_used	String	Indicates the usage status. 1: in use; 0: idle.
rule_ids	String	ID of a rule in a rule set.
is_default	String	Indicates whether the default rule set of the language is used. 0: no; 1: yes
is_devcloud_project_default	String	Indicates whether the rule set is the default rule set of the language in the project. 0: no; 1: yes.
is_default_template	String	Indicates whether the rule set is a system rule set. 0: no; 1: yes.

Status code: 400

Table 4-169 Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Status code: 401

Table 4-170 Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Example Requests

```
GET https://{endpoint}/v2/{project_id}/rulesets
```

Example Responses

Status code: 200

Request succeeded!

```
{
  "info" : [ {
    "template_id" : "9698e0cb9d2441c3bd8cec5f8641696e",
    "language" : "Java",
    "template_name" : "test1111",
    "creator_id" : "ae161856f2604229ae12056478a7919f",
    "creator_name" : "Mr. Yang",
    "template_create_time" : "2021-10-21T02:14:55Z",
    "is_used" : "0",
    "rule_ids" : "",
    "is_default" : "0",
    "is_default_template" : "1",
    "is_devcloud_project_default" : "0"
  }, {
    "template_id" : "9698e0cb9d2441c3bd8cec5f8641696e",
    "language" : "cpp",
    "template_name" : "test222",
    "creator_id" : "ae161856f2604229ae12056478a7919f",
    "creator_name" : "Mr. Yang",
    "template_create_time" : "2021-10-21T02:14:55Z",
    "is_used" : "1",
    "rule_ids" : "11152,11153,11154,11155",
    "is_default" : "0",
    "is_default_template" : "1",
    "is_devcloud_project_default" : "1"
  } ],
  "total" : 2
}
```

Status code: 400

Bad Request

```
{
  "error_code" : "CC.00000000",
  "error_msg" : "The network is busy. Please try again later."
}
```

Status code: 401

Unauthorized

```
{
  "error_code" : "CC.00000003",
  "error_msg" : "The authentication information has expired."
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request
401	Unauthorized

Error Codes

See [Error Codes](#).

4.3.4 Viewing the Rule List of a Rule Set

Function

This API is used to query the rule list based on conditions such as the project ID and rule set ID.

URI

GET /v2/{project_id}/ruleset/{ruleset_id}/rules

Table 4-171 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
ruleset_id	Yes	String	Rule Set ID

Table 4-172 Query Parameters

Parameter	Mandatory	Type	Description
types	Yes	String	Status '1: query all; 2: enabled; 3: disabled' Default: 1
languages	No	String	Rule Language
tags	No	String	Tag to which the rule belongs
offset	No	Integer	Pagination index, offset Minimum: 0 Default: 0
limit	No	Integer	Number of records displayed on each page. Minimum: 1 Maximum: 1000 Default: 100

Request Parameters

Table 4-173 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	User token. It can be obtained by calling an IAM API. The value of X-Subject-Token in the response header is the user token. Minimum: 1 Maximum: 100000

Response Parameters

Status code: 200

Table 4-174 Response body parameters

Parameter	Type	Description
info	Array of RuleItem objects	Rule list of a rule set.
total	Integer	Total

Table 4-175 RuleItem

Parameter	Type	Description
rule_id	String	Rule ID
rule_language	String	Rule Language
rule_name	String	Rule Name
rule_severity	String	Rule Problem Severity
tags	String	Rule Tag
checked	String	Rule status. The options are as follows: 0: disabled; 1: enabled.
rule_config_list	Array of RuleConfig objects	Threshold information about rule configuration parameters

Table 4-176 RuleConfig

Parameter	Type	Description
id	Integer	Rule Configuration ID
rule_id	Integer	Rule ID.
default_value	String	Default Value
option_value	String	Before acceleration
option_key	String	Key of the current rule configuration item.
option_name	String	Name of the current rule configuration item
template_id	String	Rule Set ID
description	String	Description

Status code: 400**Table 4-177** Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Status code: 401**Table 4-178** Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Example Requests

GET https://{endpoint}/v2/{project_id}/ruleset/{ruleset_id}/rules

Example Responses

Status code: 200

Request succeeded!

```
{  
  "info" : [ {  
    "rule_language" : "cpp",
```

```
"rule_id" : "1614",
"checked" : "1",
"rule_name" : "A buffer must have size of 2 integers if used as parameter of pipe().",
"rule_severity" : "1",
"tags" : "cwe",
"rule_config_list" : [ ]
}, {
"rule_language" : "cpp",
"rule_id" : "1611",
"checked" : "1",
"rule_name" : "A buffer must have size of 2 integers if used as parameter of pipe().",
"rule_severity" : "1",
"tags" : "cwe",
"rule_config_list" : [ {
"rule_id" : 250,
"rule_id" : 11707,
"template_id" : "906e7eac47dd4bde9c984f5e6f2a54e8",
"option_key" : "threshold",
"option_name" : "Adding a Threshold",
"option_value" : "10",
"default_value" : "20",
"description" : "If the code complexity exceeds the threshold, the code is identified as an ultra-large cyclomatic complexity. The difference between the cyclomatic complexity statistics and the cyclomatic complexity statistics is that only the number of switches is counted, and the case and default values are not counted. This parameter is mandatory."
}, {
"rule_id" : 251,
"rule_id" : 11707,
"template_id" : "906e7eac47dd4bde9c984f5e6f2a54e8",
"option_key" : "threshold_modify",
"option_name" : "Modifying a threshold",
"option_value" : "30",
"default_value" : "20",
"description" : "If the code complexity exceeds the threshold, the code is identified as ultra-large cyclomatic complexity. The difference between the cyclomatic complexity statistics and cyclomatic complexity statistics is that only the number of switches is counted, and the case and default values are not counted. The modified threshold must be greater than the new threshold. This parameter is optional."
}
]
}, {
"total" : 2
}
}
```

Status code: 400

Bad Request

```
{
"error_code" : "CC.00000000",
"error_msg" : "The network is busy. Please try again later."
}
```

Status code: 401

Unauthorized

```
{
"error_code" : "CC.00000003",
"error_msg" : "The authentication information has expired."
}
```

Status Codes

Status Code	Description
200	Request succeeded!

Status Code	Description
400	Bad Request
401	Unauthorized

Error Codes

See [Error Codes](#).

4.3.5 Deleting a User-Defined Rule Set

Function

This interface is used to delete a user-defined rule set. The rule set in use or the default rule set cannot be deleted.

URI

DELETE /v2/{project_id}/ruleset/{ruleset_id}

Table 4-179 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
ruleset_id	Yes	String	Rule Set ID

Request Parameters

Table 4-180 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	Specifies the user token. It can be obtained by calling an IAM API. The value of X-Subject-Token in the response header is the user token. Minimum: 1 Maximum: 100000

Response Parameters

Status code: 400

Table 4-181 Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Status code: 401**Table 4-182** Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Example Requests

```
DELETE https://{endpoint}/v2/{project_id}/ruleset/{ruleset_id}
```

Example Responses

Status code: 200

Request succeeded!

```
null
```

Status code: 400

Bad Request

```
{  
  "error_code" : "CC.00000000",  
  "error_msg" : "The network is busy. Please try again later."  
}
```

Status code: 401

Unauthorized

```
{  
  "error_code" : "CC.00000003",  
  "error_msg" : "The authentication information has expired."  
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request

Status Code	Description
401	Unauthorized

Error Codes

See [Error Codes](#).

4.3.6 Set the default rule set configuration for the language corresponding to each project.

Function

Set the default rule set configuration for the language corresponding to each project.

URI

POST /v2/{project_id}/ruleset/{ruleset_id}/{language}/default

Table 4-183 Path Parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID
ruleset_id	Yes	String	Rule Set ID
language	Yes	String	Rule Set Language

Request Parameters

Table 4-184 Request header parameters

Parameter	Mandatory	Type	Description
X-Auth-Token	Yes	String	It can be obtained by calling an IAM API. The value of X-Subject-Token in the response header is the user token. Minimum: 1 Maximum: 100000

Response Parameters

Status code: 400

Table 4-185 Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Status code: 401**Table 4-186** Response body parameters

Parameter	Type	Description
error_code	String	Error Codes
error_msg	String	Description

Example Requests

```
POST https://{endpoint}/v2/{project_id}/{language}/ruleset/{ruleset_id}/default
```

Example Responses

Status code: 200

Request succeeded!

```
null
```

Status code: 400

Bad Request

```
{  
  "error_code" : "CC.00000000",  
  "error_msg" : "The network is busy. Please try again later."  
}
```

Status code: 401

Unauthorized

```
{  
  "error_code" : "CC.00000003",  
  "error_msg" : "The authentication information has expired."  
}
```

Status Codes

Status Code	Description
200	Request succeeded!
400	Bad Request

Status Code	Description
401	Unauthorized

Error Codes

See [Error Codes](#).

5 Application Examples

[5.1 Example 1: Querying the Task List](#)

[5.2 Example 2: Querying the Defect Summary](#)

5.1 Example 1: Querying the Task List

Scenario

Query the task list of a project based on **DEVCLLOUD_PROJECT_UUID**.

Constraints

None

Involved API

The following API is involved:

Querying the Task List: queries the task list based on **DEVCLLOUD_PROJECT_UUID**.

Procedure

Query the task list.

- API information

URI format: **GET /v2/{project_id}/tasks**

- Request example

GET https://{endpoint}/v2/2b31ed520xxxxxxeb6e57xxxxxxx/tasks

Obtain the information from [1.3 Endpoints](#).

- Example response

```
{
  "result": [
    {
      "projectId": "b9db371361724226937f3280df1d4fc4",
      "devCloudProjectId": "a2b9a42ee6494e0794f5655f8535920e",
      "devCloudProjectName": "TestDemo",
      "codeHubName": "portal-ts",
```

```
"codeHubWebUrl":"https://xxxx/portal-ts",
"codeLineTotal":"63371",
"projectName":"portal-ts-abctest111222",
"creatorId":"ae161856f2604229ae12056478a7919f",
"creatorName":"devcloud_devcloud_y00336947_01",
"domainId":"78d0e09c1ad0425b9b4f3a8fdd066164",
"domainName":"devcloud_devcloud_y00336947_01",
"gitUrl":"git@xxxx/portal-ts.git",
"branch":"master",
"language":["
\"html\"
],
"filePath":["
\"\"
],
"lastCheckTime\":"2019-09-24 15:00:41",
"codeQuality":"141.7",
"codeLine":"55777",
"projectStatus":"2",
"riskCoefficient":"7905",
"starLevel":"0",
"defectInfo":{"
"totalCount":"0",
"criticalCount":"0",
"seriousCount":"0",
"normalCount":"0",
"promptCount":"0"
},
"defectStatusInfo":{"
"totalCount":"2952",
"newCount":"2952",
"ignoredCount":"0",
"fixedCount":"0"
},
"progress":{
},
"createdAt":"2019-08-08 17:56:40",
"duplications":"24.0%",
"complexityNum":"0",
"complexityAvg":"0",
"duplicatedLines":"15206",
"commentLines":"0",
"commentRatio":"0",
"duplicatedBlocks":"0",
"lastExecTime":"2 minutes 4 seconds",
"checkType":"source",
"dependenceType":"",
"logDatasStr":{"
}
},
{
"projectId":"b9db371361724226937f3280df1d4fc4",
"devCloudProjectId":"a2b9a42ee6494e0794f5655f8535920e",
"devCloudProjectName":"TestDemo",
"codeHubName":"portal-ts",
"codeHubWebUrl":"https://xxxx/portal-ts",
"codeLineTotal":"63371",
"projectName":"portal-ts-abctest111222",
"creatorId":"ae161856f2604229ae12056478a7919f",
"creatorName":"devcloud_devcloud_y00336947_01",
"domainId":"78d0e09c1ad0425b9b4f3a8fdd066164",
"domainName":"devcloud_devcloud_y00336947_01",
"gitUrl":"git@xxxx/portal-ts.git",
"branch":"master",
"language":["
\"html\"
],

```

```
"filePath":["
  "\
"],
"lastCheckTime\":"2019-09-24 15:00:41",
"codeQuality":"141.7",
"codeLine":"55777",
"projectStatus":"2",
"riskCoefficient":"7905",
"starLevel":"0",
"defectInfo":{
  "totalCount":"0",
  "criticalCount":"0",
  "seriousCount":"0",
  "normalCount":"0",
  "promptCount":"0"
},
"defectStatusInfo":{
  "totalCount":"2952",
  "newCount":"2952",
  "ignoredCount":"0",
  "fixedCount":"0"
},
"progress":{
}
},
"createdAt":"2019-08-08 17:56:40",
"duplications":"24.0%",
"complexityNum":"0",
"complexityAvg":"0",
"duplicatedLines":"15206",
"commentLines":"0",
"commentRatio":"0",
"duplicatedBlocks":"0",
"lastExecTime":"2 minutes 4 seconds",
"checkType":"source",
"dependenceType":"",
"logDatasStr":{
}
}
],
"status":"success"
}
```

5.2 Example 2: Querying the Defect Summary

Scenario

Query the defect summary based on the check task ID, including the issue overview, issue status, cyclomatic complexity, and code repetition rate.

Constraints

The task exists.

Involved API

The following API is involved:

Querying the Defect Summary: queries the defect summary, including the issue overview, issue status, cyclomatic complexity, and code repetition rate.

Procedure

- API information
URI format: **GET /v2/tasks/{task_id}/defects-summary**
- Request example
GET https://{endpoint}/v2/tasks/2b31ed520xxxxxebedb6e57xxxxxxx/defects-summary
Obtain the information from [1.3 Endpoints](#).

- Example response

```
{
  "result":{
    "info":{
      "projectId":"b9db371361724226937f3280df1d4fc4",
      "devCloudProjectId":"a2b9a42ee6494e0794f5655f8535920e",
      "devCloudProjectName":"TestDemo",
      "codeHubName":"portal-ts",
      "codeHubWebUrl":"https://xxxxx/portal-ts",
      "codeLineTotal":"63371",
      "projectName":"portal-ts-abctest111222",
      "creatorId":"ae161856f2604229ae12056478a7919f",
      "creatorName":"devcloud_devcloud_y00336947_01",
      "domainId":"78d0e09c1ad0425b9b4f3a8fdd066164",
      "domainName":"devcloud_devcloud_y00336947_01",
      "gitUrl":"git@xxxxx/portal-ts.git",
      "branch":"master",
      "language":["
        \"html\"
      ],
      "filePath":["
        \"\"
      ],
      "lastCheckTime\":\"2019-09-24 15:00:41",
      "codeQuality":"141.7",
      "codeLine":"55777",
      "projectStatus":"2",
      "riskCoefficient":"790",
      "starLevel":"0",
      "defectInfo":{
        "totalCount":"0",
        "criticalCount":"0",
        "seriousCount":"0",
        "normalCount":"0",
        "promptCount":"0"
      },
      "defectStatusInfo":{
        "totalCount":"2952",
        "newCount":"2952",
        "ignoredCount":"0",
        "fixedCount":"0"
      },
      "progress":{
      },
      "createdAt":"2019-08-08 17:56:40",
      "duplications":"24.0%",
      "complexityNum":"0",
      "complexityAvg":"0",
      "duplicatedLines":"15206",
      "commentLines":"0",
      "commentRatio":"0",
      "duplicatedBlocks":"0",
      "lastExecTime":"2 minutes 4 seconds",
      "checkType":"source",
      "dependenceType":"",
      "logDatasStr":{
      }
    }
  }
}
```

```
}  
},  
"status": "success"  
}
```

6 Appendixes

[6.1 Status Codes](#)

[6.2 Error Codes](#)

[6.3 Obtaining a Project ID](#)

[6.4 Obtaining an Account ID](#)

6.1 Status Codes

Table 6-1 describes status codes.

Table 6-1 Status codes

Status Code	Message	Description
100	Continue	Continue sending requests. This temporary response is used to inform the client that some requests have been received and not rejected by the server.
101	Switching Protocols	Switch protocols. A protocol can only be switched to a more advanced protocol. For example, switch to a later version of HTTP.
201	Created	The request for creating a resource has been implemented.
202	Accepted	The request has been accepted, but has not finished being processed yet.
203	Non-Authoritative Information	Unauthorized information. The request is successful.

Status Code	Message	Description
204	NoContent	The request has been successfully implemented, but the HTTP response does not contain a response body. This status code is returned in response to an HTTP OPTIONS request.
205	Reset Content	The server has successfully processed the request after resetting the content.
206	Partial Content	The server has successfully processed a part of the GET request.
300	Multiple Choices	There are multiple choices for a requested resource. A list of resource characteristics and addresses is returned for the client such as a browser to choose from.
301	Moved Permanently	A requested resource has been permanently moved to the given URI indicated in the response.
302	Found	The requested resource has been temporarily moved.
303	See Other	Retrieve another URL, using a GET or POST method.
304	Not Modified	The requested resource has not been modified. When the server returns this status code, no resource is returned.
305	Use Proxy	The requested resource is available only through a proxy.
306	Unused	This HTTP status code is no longer used.
400	BadRequest	Invalid request. The client should modify the request instead of re-initiating it.
401	Unauthorized	The authorization information provided by the client is incorrect or invalid.
402	Payment Required	Reserved request.
403	Forbidden	Request rejected. The server has received the request and understood it, but refuses to respond to it. The client should not repeat the request without modifications.
404	NotFound	The requested resource cannot be found. The client should modify the request instead of re-initiating it.

Status Code	Message	Description
405	MethodNotAllowed	A request method is not supported for the requested resource. The client should modify the request instead of re-initiating it.
406	Not Acceptable	The server cannot implement the request based on the content characteristics of the request.
407	Proxy Authentication Required	This code is similar to 401, but indicates that the client must first authenticate itself with the proxy.
408	Request Time-out	The server timed out waiting for the request. The client may re-initiate the request without modifications at any time later.
409	Conflict	The request cannot be processed due to a conflict. This status code indicates that the resource that the client attempts to create already exists, or the request fails to be processed because of the update of the conflict request.
410	Gone	The requested resource cannot be found. This status code indicates that the requested resource has been deleted permanently.
411	Length Required	The server fails to process the request which does not contain the Content-Length header field.
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 temporarily cannot process the request, the response will contain a Retry-After header field.
414	Request-URI Too Large	The request URI is too long for the server to process.
415	Unsupported Media Type	The server does not support the media type 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.

Status Code	Message	Description
422	UnprocessableEntity	The request is well-formed but is unable to respond due to semantic errors.
429	TooManyRequests	The client sends too many requests to the server within a given time, exceeding the client's access frequency limit or beyond the server's processing capability. In this case, the client should retry after the time period specified in the Retry-After response header.
500	InternalServerError	The server is able to receive the request but it cannot understand the request.
501	Not Implemented	The server does not support the requested function, and therefore cannot implement the request.
502	Bad Gateway	The server acting as a gateway or proxy receives an invalid response from a remote server.
503	ServiceUnavailable	The requested service is invalid. The client should modify the request instead of re-initiating it.
504	ServerTimeout	The server could not return a timely response. The response will reach the client only if the request carries a timeout parameter.
505	HTTP Version not supported	The server does not support the HTTP protocol version used in the request.

6.2 Error Codes

Since no result data is returned in the case of an API call error, you can identify the error cause based on the error codes of each API.

If HTTP status code 400 or 505 is returned after an API call error, you can find the specific error code and description in the response body.

If you are unable to identify the cause of an error, contact customer service and provide the error code for quick troubleshooting.

Format of an Error Response Body

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

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

In the preceding information, **error_code** is an error code, and **error_msg** describes the error.

Error Code Description

If an error code starting with **APIGW** is returned after you call an API, rectify the fault by referring to the instructions provided in [API Gateway Error Codes](#).

Table 6-2 Error codes

Status Code	Error Code	Error Message	Description	Solution
200	CC.00050000	parameters not valid	Invalid parameters.	Check the parameter formats.
200	CC.00050001	query task list error	Querying the task list throws an error.	Contact customer service.
200	CC.00050002	query task detail error	Querying task details throws an error.	Contact customer service.
200	CC.00050003	query templates error	Viewing the task configuration template throws an error.	Contact customer service.
200	CC.00050004	query task history error	Querying the task history throws an error.	Contact customer service.
200	CC.00050005	create task error	Creating a task throws an error.	Contact customer service.
200	CC.00050006	service inner error	Internal service error.	Contact customer service.
200	CC.00050007	IAM authentication failed	IAM authentication failed.	Check whether the authentication token is valid.
200	CC.00050010	current user does not have permission	Insufficient permission.	Check the permission.

Status Code	Error Code	Error Message	Description	Solution
200	CC.00050011	not in any project	The current user is not a member of any project of the tenant.	Join the project.
200	CC.00050012	retry later	The network is busy.	Try again later or contact customer service.
200	CC.00050013	project not exist	The project does not exist or has been deleted.	Modify project-related parameters.
200	CC.00050014	call getTaskBasicInfo failed	Failed to call the getTaskBasicInfo API.	Try again later or contact customer service.
200	CC.00050015	task is running	The task is being executed. Try again later.	Try again later.
200	CC.00050016	create task failed	An error occurred when creating a task. Creation failed.	Check related parameters or contact customer service.
200	CC.00050017	call updateIssueStatus failed	Ignore the warning in the latest scanning result.	Ignore the warning in the latest scanning result.
200	CC.00050019	operations that are not supported by domain of the earlier version	Operation not supported by tenants using earlier versions	Operation not supported by tenants using earlier versions
200	CC.00050020	The current task has not been checked or has been checked successfully. Please check or try again after the check is successful	The current task has not been checked or checked successfully. Check or try again after the check is successful.	Check or try again after the check is successful.

6.3 Obtaining a Project ID

Obtaining a Project ID by Calling an API

You can obtain the project ID by calling the API used to [query project information](#).

The API for obtaining a project ID is **GET [https://{Endpoint}/v3/projects/{Endpoint}](#)**. *{Endpoint}* indicates the endpoint of IAM, which can be obtained from [1.3 Endpoints](#). For details about API authentication, see [Authentication](#).

In the following example, **id** indicates the project ID.

```
{
  "projects": [
    {
      "domain_id": "65382450e8f64ac0870cd180d14e684b",
      "is_domain": false,
      "parent_id": "65382450e8f64ac0870cd180d14e684b",

      "name": "eu-west-101",
      "description": "",
      "links": {
        "next": null,
        "previous": null,
        "self": "https://www.example.com/v3/projects/a4a5d4098fb4474fa22cd05f897d6b99"
      },
      "id": "a4a5d4098fb4474fa22cd05f897d6b99",
      "enabled": true
    }
  ],
  "links": {
    "next": null,
    "previous": null,
    "self": "https://www.example.com/v3/projects"
  }
}
```

Obtaining a Project ID from the Console

A project ID is required for some URLs when an API is called. To obtain a project ID, perform the following operations:

1. Log in to the management console.
2. Hover the mouse pointer over the username in the upper right corner and choose **My Credentials** from the drop-down list.

Choose **My Credentials** > **API Credentials** to view the project ID in the project list.

6.4 Obtaining an Account ID

An account ID is required for some URLs when an API is called. To obtain an account ID, perform the following operations:

1. Log in to the management console.

2. Hover the mouse pointer over the username in the upper right corner and choose **My Credentials** from the drop-down list.
Choose **My Credentials > API Credentials** to view the account ID.

7 Change History

Changes between document issues are cumulative. The latest document issue contains all the changes made in earlier issues.

Description	Release Date
This issue is the first official release.	2023-11-30