Mobile Security

API Reference

API Reference

API overview

API	Type	Description	Maximum Calls
Vulnerability scan API	Paid	- Vulnerability scan interface - Vulnerability scan result query interface	50 times per day
APP hardening API	Paid	- APP hardening interface - APP hardening result query interface	50 times per day

Call method

Request structure

Service address

The access address for the Mobile Security API is jaq.aliyuncs.com.

Communication protocol

The Mobile Security only supports **HTTPS** communication.

Request method

The Mobile Security supports **HTTPS GET** request. The request parameters must be described in the request URL.

Request parameters

Each request must specify the operation to be performed (for example, Authenticate) through the action parameters. The action parameters include both public request parameters mandatory for each operation and request parameters required for specified operation.

Character encoding

Requests and returned results are encoded using the **UTF-8** character set.

Public parameters

Public request parameters

Public request parameters are the request parameters used in each API.

Name	Type	Required?	Description
Format	String	No	Type of the value returned. JSON and XML are supported. The default value is XML.
Version	String	Yes	API version, in the format of YYYY-MM-DD. The current version is 2014-05-26.
AccessKeyId	String	Yes	Identity of the key that Alibaba Cloud issued to a user to access services.
Signature	String	Yes	Signature result. For details about signature calculation methods, refer to Signature mechanism.

SignatureMethod	String	Yes	Signature method. HMAC-SHA1 is supported currently.
Timestamp	String	Yes	Request time stamp. The date and time stamp uses the UTC system and the format complies with ISO8601. The format is YYYY-MM-DDThh:mm:ssZ. For example, 2014-05-26T12:00:00Z (indicates 20:00:00, May 26, 2014, Beijing time).
SignatureVersion	String	Yes	Signature algorithm version. The current version is 1.0.
SignatureNonce	String	Yes	Unique random number, which is used to prevent network replay attacks. Different random numbers must be used for different requests.
Action	String	Yes	API name, for example Shield.

Request example

```
https://jaq.aliyuncs.com/
?Format=XML&Version=2016-04-12
&Action=Shield&AccessKeyId=keyId
&Signature=5Ag6fYxyiVJbHfuJdQFZPDyso40=
&SignatureMethod=HMAC-SHA1
&Timestamp=2016-06-02T15:22:59Z
&SignatureVersion=1.0
&SignatureNonce=c8233541-94c5-4a9f-abcc-425c66b6cf75
&RegionId=cn-hangzhou
……
```

Public returned parameters

Each time you send an call request to an API, the system returns errorCode and errorMsg, regardless of whether the request is successful or not.

Return example

JSON format

```
{     " ErrorCode": "xxx",
" ErrorMsg": 0,
/* Return result data */ }
```

XML format

```
<?xml version="1.0" encoding="UTF-8"?>
<!--Result root node--> <API name+Response>
……
<ErrorCode>4C467B38-3910-447D-87BC-AC049166F216</ ErrorCode >
<ErrorMsg>0</ ErrorMsg >
<!--Result root node--> </API name+Response>
```

Signature mechanism

The Mobile Security service authenticates each access request. By using Access Key ID and Access Key Secret, CloudMonitor performs symmetric encryption to authenticate the request sender.

The Access Key ID and Access Key Secret are officially issued to visitors by Alibaba Cloud (visitors can apply for and manage them at Alibaba Cloud’ s official website). The Access Key ID indicates the identity of a visitor, and the Access Key Secret is the key used to encrypt a signature string and verify it at the server end. It must be kept confidential and should only be available to Alibaba Cloud and the user.

Signing process

1. Construct a Canonicalized Query String by using request parameters.

a. Order the request parameters by parameter name alphabetically. These parameters include the public request parameters described in this document and user-defined parameters for a given request interface, but do not include the signature parameter mentioned in public request parameters.

Note: When a request is submitted using the GET method, the request parameters are the same as those included in the parameter section of the request URI. That is, the section of the URI following the “?” and connected by “&”.

b. Perform URL encoding of the name and value of each request parameter (only supports the UTF-8 character set). The URL encoding rules are as follows:

- English letters A–Z and a–z, digits 0–9, and characters “-“, “_”, “.”, and “~” are not encoded;
- Other characters are encoded in the “%XY” format, with XY representing the characters’ ASCII code in hexadecimal notation. For example, double quotes (“) are encoded into %22;
- Extended UTF-8 characters are encoded in the “%XY%ZA…” format.
- Note that a space ( ) is encoded into %20 instead of a plus sign (+).

Note: Libraries that support URL encoding (for example, java.net.URLEncoder) are typically encoded according to the rules for the “application/x-www-form-urlencoded” MIME type. If this encoding method is used, replace the plus sign (+) in the encoded string with %20, the asterisk (*) with %2A, and change %7E back to the tilde (~) to conform to the encoding rules described above.

c. Connect the encoded parameter names and values with equal signs (=) to obtain several parameter pairs.

d. Connect the parameter pairs with the “&” symbol based on the alphabetic sorting results by parameter name mentioned above to obtain a Canonicalized Query String.

2. Follow the rules below to construct the string used for signature calculation from the Canonicalized Query String obtained in the previous step.

The construction method is as follows:

- HTTPMethod indicates that the HTTP method (for example, GET) is used to submit a request.
- percentEncode(“/“) is the encoded value (“%2F”) of the character “/“, which is obtained according to the URL encoding rules described in 1.b.
- percentEncode(CanonicalizedQueryString) is the Canonicalized Query String (constructed in Step 1) that is encoded according to the URL encoding rules described in 1.b.

3. Use the above signature sting to calculate the signature’s HMAC value based on RFC2104 definitions.

Note: The key used for calculating the signature is the user’s Access Key Secret ending with the “&” character (ASCII:38). Secure Hash Algorithm 1 (SHA1) is used.

```
public static byte[] HmacSHA1(byte[] data, byte[] key) throws Exception {
try {
SecretKeySpec signingKey = new SecretKeySpec(key, "HmacSHA1");
Mac mac = Mac.getInstance("HmacSHA1");
mac.init(signingKey);
byte[] rawHmac = mac.doFinal(data);
return rawHmac;
} catch (Exception e) {
e.printStackTrace();
return null;
}
}
```

4. Encode the HMAC value into a string according to the Base64 encoding rules to obtain the

signature value.

5. Add the signature value as the Signature parameter to the request parameters. The request signing process ends.

Note: When the signature value is submitted to the OPENSEARCH server as the final request parameter value, the signature value will be URL encoded like other parameters according to the RFC3986 rules.

Example

The request for obtaining hardening results is used as an example.

The URL before signing is:

```
https://jaq.aliyuncs.com/
?AccessKeyId=testid
&Action=GetShieldResult
&Format=JSON
&ItemId=366ce1a0-8b71-4409-bfcc-961811805077
&RegionId=cn-hangzhou
&SignatureMethod=HMAC-SHA1
&SignatureNonce=c08d7277-07b9-417c-86ac-3fd03d00115d
&SignatureVersion=1.0
&Timestamp=2016-06-16T04%3A24%3A25Z
&Version=2016-04-12
```

Therefore, the StringToSign is:

```
GET
&%2F
&AccessKeyId%3Dtestid
&Action%3DGetShieldResult
&Format%3DJSON
&ItemId%3D366ce1a0-8b71-4409-bfcc-961811805077
&RegionId%3Dcn-hangzhou
&SignatureMethod%3DHMAC-SHA1
&SignatureNonce%3Dc08d7277-07b9-417c-86ac-3fd03d00115d
&SignatureVersion%3D1.0
&Timestamp%3D2016-06-16T04%253A24%253A25Z
&Version%3D2016-04-12
```

Assume that the Access Key ID is testid, the Access Key Secret is testsecret, and the key used for HMAC calculation is testsecret&.

The calculated signature value is: 22CtcegKLClHArSFXx/qqn8dUYI=.

The signed request URL is (note the added Signature parameter):

```
https://jaq.aliyuncs.com/
```

```
?AccessKeyId=testid
&Action=GetShieldResult
&Format=JSON
&ItemId=366ce1a0-8b71-4409-bfcc-961811805077
&RegionId=cn-hangzhou
&SignatureMethod=HMAC-SHA1
&SignatureNonce=c08d7277-07b9-417c-86ac-3fd03d00115d
&SignatureVersion=1.0
&Timestamp=2016-06-16T04%3A24%3A25Z
&Version=2016-04-12
&Signature=22CtcegKLClHArSFXx%2Fqqn8dUYI%3D
```

APP Hardening API

Overview

Before calling the APP hardening API, ensure that a professional edition of APP hardening service is purchased and bind to the request application. Refer to Quick Start for details.

The APP hardening API includes two types:

APP hardening interface

APP hardening result query interface

APP hardening interface

Request parameters

Name	Parent Node	Type	Required?	Description
AppInfo	-	String (JSON format)	No	Information of the application to be hardened.
dataType	AppInfo	Number	No	Application data type. Value:

				- 1: App URL
data	AppInfo	String	No	Application data. When dataType=1, fill in the download address of the application package.
md5	AppInfo	String	No	MD5 value of the application package. Required for file verification when dataType=1.
size	AppInfo	Number	No	Application package size(unit: bytes). Required for file verification when dataType=1.
callbackUrl	AppInfo	String	No	Callback address of the reverse notification when the task is completed. Required when dataType=1. This notification is a GET request, whose Request URL is: callbackUrl+"?item_id=xxx&task_status=1". Where, item_id denotes task ID returned by the APP hardening interface, and task_status denotes the status of the task. The task_status

				values include: - 1: finished - 2: processing - 3: processing error - 4: processing timeout For APP hardening, if in the received request the task_status is 1, you can view the hardening results through the corresponding query interface.
appOsType	AppInfo	Number	No	Application type. Value: - 1: apk
Channel	-	String (JSON format)	No	Channel list. Required for multi-channel hardening.
metaName	Channel	String	No	Android: Name of the meta-data label used to indicate channel information in AndroidManifest.xml.
values	Channel	List<String>	No	List of channel names.
Enhance	-	String (JSON format)	No	User-defined hardening information.

				Required when user-defined hardening is configured.
enhanceType	Enhance	Number	No	DEX hardening types. Values: - 0: no hardening - 1: lightweight hardening - 2: overall hardening
javaAntiDex2Jar	Enhance	String (JSON format)	No	Anti-dex2jar information of Java. Required when Java-layer constant hardening is configured.
percent	javaAntiDex2Jar	Number	No	Percentage configuration for anti-dex2jar.
javaConstEncrypt	Enhance	String (JSON format)	No	Information of Java-layer constant hardening. Required when Java-layer constant hardening is configured.
percent	javaConstEncrypt	Number	No	Percentage configuration for constant hardening.
soEnhance	Enhance	String (JSON format)	No	SO hardening information. Required when user-defined SO hardening

				is configured.
enhanceType	Enhance	Number	No	SO hardening types. Values: - 0: no hardening - 1: lightweight hardening - 2: heavy weight hardening
soFileList	enhanceType	String	No	List of SO files to be hardened.

Request example

```
https://jaq.aliyuncs.com/
?Format=JSON
&Channel=%7B%22metaName%22%3A%22channel%22%2C%22values%22%3A%5B%2291%22%2C%22360%22%5D%7D
&SignatureMethod=HMAC-SHA1
&Signature=EYXamTa%2BRIafYaPoUdwHdwWpSvA%3D
&Timestamp=2016-06-05T06%3A41%3A30Z
&Enhance=%7B%27enhanceType%27%3A+-1%2C+%27javaAntiDex2jar%27%3A+%7B%27percent%27%3A+20%7D%2C%27javaConstEncrypt%27%3A+%7B%27percent%27%3A+20%7D%2C%27soEnhance%27%3A+%7B%27enhanceType%27%3A+1%2C%27soFileList%27%3A+%5B%27lib%2Farmeabi%2Flibbitmaps.so%27%2C%27lib%2Farmeabi%2Flibgifimage.so%27%5D%7D%7D
&Action=DiyShield
&AccessKeyId=accessKeyId
&AppInfo=%7B%22appOsType%22%3A1%2C%22callbackUrl%22%3A%22http%3A%2F%2Faaa.com%2Fcallback%22%2C%22data%22%3A%22http%3A%2F%2Fg01.alibaba-inc.com%2Ftfscom%2FLB1PaMeKXXXXXX8XFXXXXXXXXXX.tfsprivate1446115983140-375%22%2C%22dataType%22%3A1%2C%22md5%22%3A%22ce86f08da845d0af6d9df2a958de17b0%22%2C%22size%22%3A1713656%7D
&RegionId=cn-hangzhou
&SignatureNonce=eac373bc-355f-4e9e-95fc-61cb986c8a80&Version=2016-04-12
&SignatureVersion=1.0
```

Return parameters

Name	Parent Node	Description

Data	-	Returned results.
itemId	Data	Unique ID of a task.
progress	Data	Task progress. Values: - 1: finished (you can view the processing results through the corresponding query interface). - 2: asynchronous processing in progress (you can view the processing results only after app_info.callback_url receives a reverse notification). Currently, app_info.data_type=1 indicates asynchronous processing, and the returned field value is 2.

Return example

JSON format

```
{
"Data": {
"ItemId": "adef0394-3370-4e94-82c6-07af0d15a9cd",
"Progress": 2
},
"ErrorMsg": "Success",
"ErrorCode": 0
}
```

XML format

```
<?xml version='1.0' encoding='UTF-8'?>
<ShieldResponse>
<Data>
<ItemId>e112d1ba-d058-4a96-ac1e-4b9f4986cf2e</ItemId>
<Progress>2</Progress>
</Data>
<ErrorMsg>Success</ErrorMsg>
<ErrorCode>0</ErrorCode>
```

```
</ShieldResponse>
```

APP hardening result query interface

Request parameters

Name	Type	Required?	Description
ItemId	String	Yes	Unique ID of a task.

Request example

```
https://jaq.aliyuncs.com/
?Format=JSON
&ItemId=366ce1a0-8b71-4409-bfcc-961811805077
&AccessKeyId=accessKeyId
&Action=GetShieldResult
&SignatureMethod=HMAC-SHA1
&RegionId=cn-hangzhou
&SignatureNonce=76c08851-b577-4e69-88f5-6c5a34038230
&SignatureVersion=1.0
&Version=2016-04-12
&Signature=OPaEAHEbqDZQyR6s1jcr858ulSE%3D
&Timestamp=2016-06-05T06%3A23%3A00Z
```

Returned parameters

Name	Parent Node	Description
Data	-	Returned results.
taskStatus	Data	Task status. Values: - 1: finished - 2: processing - 3: processing error - 4: processing timeout
appList	Data	Application list after hardening. Returned only when the task is completed. One file for common hardening, one file each channel for multi-channel hardening.

channel	appList	Channel name. Valid for multi-channel hardening.
appUrl	appList	Download address of the hardened application.
obfuscateResult	Data	Returned result of obfuscation rate calculation.
totalClasses	obfuscateResult	Total quantity of classes.
totalFields	obfuscateResult	Total quantity of member variables.
totalMethods	obfuscateResult	Total quantity of methods.
obfuscatedClasses	obfuscateResult	Quantity of obfuscated classes.
obfuscatedFields	obfuscateResult	Quantity of obfuscated member variables.
obfuscatedMethods	obfuscateResult	Quantity of obfuscation methods.
obfuscatedPercent	obfuscateResult	Overall obfuscation rate.

Return example

JSON format


```
{
"Data": {
"ObfuscateResult": {
"ObfuscatedClasses": 2870,
"TotalClasses": 7880,
"TotalFields": 35583,
"ObfuscatedMethods": 49852,
"ObfuscatedPercent": 32,
"ObfuscatedFields": 12291,
"TotalMethods": 49852
},
"AppList": {
"ChannelAppInfo": [
"{\"appUrl\":\"http://mobisecenhance-public-apk.cn-hangzhou.oss-pub.aliyun-inc.com/2016-04-25-apk/com.taobao.movie.android-20160425-21-50-37-103.apk.91?Expires=1462631498&OSSAccessKeyId=9gt0Lx7iSdCEWgzr&Signature=tn7kI13Oqt/Y/LA4CsYwL5g25Jw%3D\",\"channel\":\"91\"}",
"{\"appUrl\":\"http://mobisecenhance-public-apk.cn-hangzhou.oss-pub.aliyun-inc.com/2016-04-25-apk/com.taobao.movie.android-20160425-21-50-37-103.apk.360?Expires=1462631499&OSSAccessKeyId=9gt0Lx7iSdCEWgzr&Signature=3C9TRRluQNS3lHKrHdXpVtrXn4A%3D\",\"channel\":\"360\"}"
]
},
```

```
"TaskStatus": 1
},
"ErrorMsg": "Success",
"ErrorCode": 0
}
```


XML format


```
<GetShieldResultResponse>
<Data>
<ObfuscateResult>
<ObfuscatedClasses>2870</ObfuscatedClasses>
<TotalClasses>7880</TotalClasses>
<TotalFields>35583</TotalFields>
<ObfuscatedMethods>49852</ObfuscatedMethods>
<ObfuscatedPercent>32.0</ObfuscatedPercent>
<ObfuscatedFields>12291</ObfuscatedFields>
<TotalMethods>49852</TotalMethods>
</ObfuscateResult>
<AppList>
<ChannelAppInfo>{"appUrl":"http://mobisecenhance-public-apk.cn-hangzhou.oss-pub.aliyun-inc.com/2016-04-25-apk/com.taobao.movie.android-20160425-21-50-37-103.apk.91?Expires=1462631498&OSSAccessKeyId=9gt0Lx7iSdCEWgzr&Signature=tn7kI13Oqt/Y/LA4CsYwL5g25Jw%3D","channel":"91"}
</ChannelAppInfo>
<ChannelAppInfo>{"appUrl":"http://mobisecenhance-public-apk.cn-hangzhou.oss-pub.aliyun-inc.com/2016-04-25-apk/com.taobao.movie.android-20160425-21-50-37-103.apk.360?Expires=1462631499&OSSAccessKeyId=9gt0Lx7iSdCEWgzr&Signature=3C9TRRluQNS3lHKrHdXpVtrXn4A%3D","channel":"360"}
</ChannelAppInfo>
</AppList>
<TaskStatus>1</TaskStatus>
</Data>
<ErrorMsg>Success</ErrorMsg>
<ErrorCode>0</ErrorCode>
</GetShieldResultResponse>
```


Vulnerability Scan API

Overview

Before calling the vulnerability scan API, ensure that a professional edition of vulnerability scan service is purchased and bind to the request application. Refer to Quick Start for details.

The vulnerability scan API includes two types:

Vulnerability scan interface

Vulnerability scan result query interface

Vulnerability scan interface

Request parameters

Name	Parent Node	Type	Required?	Description
AppInfo	-	String (JSON format)	Yes	Information of the application to be scanned.
dataType	AppInfo	Number	No	Application data type. Values: - 1: APP URL - 2: APP MD5
data	AppInfo	String	No	Application data. - When dataType=1, fill in the download address of the application package. - When dataTy

				pe=2, fill in the MD5 value of the application package.
md5	AppInfo	String	No	MD5 value of the application package. Required for file verification when dataType=1.
size	AppInfo	Number	No	Application package size(unit: bytes). Required for file verification when dataType=1.
callbackUrl	AppInfo	String	No	Callback address of the reverse notification when the task is completed. Required when dataType=1. This notification is a GET request, whose Request URL is: callbackUrl+"?item_id=xxx&task_status=1". Where, item_id denotes the task ID returned by the vulnerability scan interface, and task_status denotes the status of the task. The

				task_status values include: - 1: finished - 2: processing - 3: processing error - 4: processing timeout When task_status is 1, 3, or 4, you can view the processing results through corresponding query interface. But the results of failed scans are not included.
appOsType	AppInfo	Number	No	Application type. Values: - 1: apk - 2: ipa (not supported currently)
ExtParam	-	String	No	Additional information. Determined upon specific service.

Request example

```
https://jaq.aliyuncs.com/
?Format=JSON
&AccessKeyId=accessKeyId
&Action=ScanVuln
&SignatureMethod=HMAC-SHA1&ExtParam=xxx
&RegionId=cn-hangzhou
&AppInfo=%7B%22appOsType%22%3A1%2C%22callbackUrl%22%3A%22http%3A%2F%2Faaa.com%2Fcallback%22%2C%22data%22%3A%22http%3A%2F%2Fg01.alibaba-inc.com%2Ftfscom%2FLB1PaMeKXXXXXX8XFXXXXXXXXXX.tfsprivate1446115983140-375%22%2C%22dataType%22%3A1%2C%22md5%22%3A%22ce86f08da845d0af6d9df2a958de17b0%22%2C%22size%22%3A1713656%7D
&SignatureNonce=4a733057-2adc-4f7f-b530-fd73fb6ad079
&SignatureVersion=1.0
&Version=2016-04-12
&Signature=sVQVNw38rOCJdn6Nq8YN6CT9jTg%3D
&Timestamp=2016-06-05T07%3A16%3A04Z
```

Returned parameters

Name	Parent Node	Description
Data	-	Returned results.
itemId	Data	Unique ID of a task.
progress	Data	Task progress. Values: - 1: finished (you can view the processing results through the corresponding query interface). - 2: asynchronous processing in progress (you can view the processing results only after app_info.callback_url receives a reverse notification). Currently, - app_info.data_type=1 indicates asynchronous processing, and the returned field value

		is 2. - app_info.data_type=2 indicates synchronous processing, and the returned field value is 1.

Return example

JSON format

```
{
"Data": {
"ItemId": "adef0394-3370-4e94-82c6-07af0d15a9cd",
"Progress": 2
},
"ErrorMsg": "Success",
"ErrorCode": 0
}
```

XML format

```
<?xml version='1.0' encoding='UTF-8'?>
<ShieldResponse>
<Data>
<ItemId>
e112d1ba-d058-4a96-ac1e-4b9f4986cf2e
</ItemId>
<Progress>2</Progress>
</Data>
<ErrorMsg>Success</ErrorMsg>
<ErrorCode>0</ErrorCode>
</ShieldResponse>
```

Vulnerability scan result query interface

Request parameters

Name	Type	Required?	Description
ItemId	String	Yes	Unique ID of a task.

Language	String	Yes	Localized language information. Refer to ISO-639 for language code, for example zh, en.
Country	String	Yes	Localized language information. Refer to ISO-3166 for country code, for example CN, US.

Request example

```
https://jaq.aliyuncs.com/
?ItemId=b888cd36-a5e4-4103-85d3-5ab0c330a5d3&Format=JSON
&SignatureMethod=HMAC-SHA1
&Country=CN
&Signature=v1EGLkqoOgFmxNCO3DjwQ6qVnpw%3D
&Timestamp=2016-06-05T07%3A50%3A08Z
&Action=GetRiskDetail
&AccessKeyId=accessKeyId
&RegionId=cn-hangzhou
&Language=zh
&SignatureNonce=d98527ea-fe68-4536-aa78-01e5b9e480aa
&Version=2016-04-12
&SignatureVersion=1.0
```

Returned parameters

Name	Parent Node	Description
Data	-	Returned results.
taskStatus	Data	Task status. Values: - 1: finished - 2: processing - 3: processing error - 4: processing timeout
vuln_info	Data	Vulnerability information.
status	vuln_info	Sub-task status. Values: - 1: finished - 2: processing - 3: processing error - 4: processing timeout

vuln_count	vuln_info	Quantity of vulnerabilities. Returned only when the task is completed.
total	vuln_count	Total quantity of vulnerabilities.
highLevel	vuln_count	Quantity of high-risk vulnerabilities.
lowLevel	vuln_count	Quantity of low-risk vulnerabilities.
midLevel	vuln_count	Quantity of medium-risk vulnerabilities.
vulnDetails	vuln_info	List of vulnerability details. Returned only when the task is completed.
name	vulnDetails	Vulnerability name.
level	vulnDetails	Vulnerability risk level. The values include high, medium, and low.
count	vulnDetails	Quantity of vulnerabilities.
locations	vulnDetails	Vulnerability location.
referenctLink	vulnDetails	Link to detailed vulnerability description.
recommendation	vulnDetails	Remediation advice.

Return example

JSON format


```
{
" Data":{
"task_status":1,
"vuln_info":{
"status":1,
"vuln_count":{
"total":20,
"high_level":10,
"mid_level":6,
"low_level":4,
"red_line":7
},
"vuln_details":{
"vuln_detail":[
{
"name": "any file-related read/write vulnerability",
"level":"high",
```

```
"count":2,
"locations":{
"string":[
"Lcom\/alibaba\/wireless\/jaq\/vulnerabilities\/VulnOpenFileMode#openFileWriteMode",
"Lcom\/alibaba\/wireless\/jaq\/vulnerabilities\/VulnOpenFileMode#openFileReadMode"
]
},
"red_line":false,
"referenct_link":"http:\/\/jaq.alibaba.com\/blog.htm?id=58",
"recommendation": "Do not use MODE_WORLD_READABLE and MODE_WORLD_WRITABLE"
}
]
}
}
}
}
```


XML format


```
<GetRiskDetailResponse>
<Data>
<task_status>1</task_status>
<vuln_info>
<status>1</status>
<vuln_count>
<total>20</total>
<high_level>10</high_level>
<mid_level>6</mid_level>
<low_level>4</low_level>
<red_line>7</red_line>
</vuln_count>
<vuln_details>
<vuln_detail>
<name>Any file-related read/write vulnerability</name>
<level>high</level>
<count>2</count>
<locations>
<string>Lcom/alibaba/wireless/jaq/vulnerabilities/VulnOpenFileMode#openFileWriteMode</string>
<string>Lcom/alibaba/wireless/jaq/vulnerabilities/VulnOpenFileMode#openFileReadMode</string>
</locations>
<red_line>false</red_line>
<referenct_link>http://jaq.alibaba.com/blog.htm?id=58</referenct_link>
<recommendation>Do not use MODE_WORLD_READABLE and MODE_WORLD_WRITABLE</recommendation>
</vuln_detail>
</vuln_details>
</vuln_info>
</result>
</GetRiskDetailResponse>
```


Error codes

Error Code	Type	Error Message
400	Parameter error	Refer to error message.

500	Internal system error	Refer to error message.