阿里云
内容安全

API参考（配置管理）

文档版本：20200414

法律声明

阿里云提醒您在阅读或使用本文档之前仔细阅读、充分理解本法律声明各条款的内容。如果您阅读或使用本文档，您的阅读或使用行为将被视为对本声明全部内容的认可。

1. 您应当通过阿里云网站或阿里云提供的其他授权通道下载、获取本文档，且仅能用于自身的合法合规的业务活动。本文档的内容视为阿里云的保密信息，您应当严格遵守保密义务；未经阿里云事先书面同意，您不得向任何第三方披露本手册内容或提供给任何第三方使用。
2. 未经阿里云事先书面许可，任何单位、公司或个人不得擅自摘抄、翻译、复制本文档内容的部分或全部，不得以任何方式或途径进行传播和宣传。
3. 由于产品版本升级、调整或其他原因，本文档内容有可能变更。阿里云保留在没有任何通知或者提示下对本文档的内容进行修改的权利，并在阿里云授权通道中不时发布更新后的用户文档。您应当实时关注用户文档的版本变更并通过阿里云授权渠道下载、获取最新版的用户文档。
4. 本文档仅作为用户使用阿里云产品及服务的参考性指引，阿里云以产品及服务的“现状”、“有缺陷”和“当前功能”的状态提供本文档。阿里云在现有技术的基础上尽最大努力提供相应的介绍及操作指引，但阿里云在此明确声明对本文档内容的准确性、完整性、适用性、可靠性等不作任何明示或暗示的保证。任何单位、公司或个人因为下载、使用或信赖本文档而发生任何差错或经济损失的，阿里云不承担任何法律责任。在任何情况下，阿里云均不对任何间接性、后果性、惩戒性、偶然性、特殊性或刑罚性的损害，包括用户使用或信赖本文档而遭受的利润损失，承担责任（即使阿里云已被告知该等损失的可能性）。
5. 阿里云文档中所有内容，包括但不限于图片、架构设计、页面布局、文字描述，均由阿里云和/或其关联公司依法拥有其知识产权，包括但不限于商标权、专利权、著作权、商业秘密等。非经阿里云和/或其关联公司书面同意，任何人不得擅自使用、修改、复制、公开传播、改变、散布、发行或公开发表阿里云网站、产品程序或内容。此外，未经阿里云事先书面同意，任何人不得为了任何营销、广告、促销或其他目的使用、公布或复制阿里云的名称（包括但不限于单独为或以组合形式包含“阿里云”、“Aliyun”、“万网”等阿里云和/或其关联公司品牌，上述品牌的附属标志及图案或任何类似公司名称、商号、商标、产品或服务名称、域名、图案标示、标志、标识或通过特定描述使第三方能够识别阿里云和/或其关联公司）。
6. 如若发现本文档存在任何错误，请与阿里云取得直接联系。

通用约定

格式	说明	样例
	该类警示信息将导致系统重大变更甚至故障，或者导致人身伤害等结果。	 **禁止：** 重置操作将丢失用户配置数据。
	该类警示信息可能会导致系统重大变更甚至故障，或者导致人身伤害等结果。	 **警告：** 重启操作将导致业务中断，恢复业务时间约十分钟。
	用于警示信息、补充说明等，是用户必须了解的内容。	 **注意：** 权重设置为0，该服务器不会再接受新请求。
	用于补充说明、最佳实践、窍门等，不是用户必须了解的内容。	 **说明：** 您也可以通过按Ctrl + A选中全部文件。
>	多级菜单递进。	单击**设置** > **网络** > **设置网络类型**。
粗体	表示按键、菜单、页面名称等UI元素。	在**结果确认**页面，单击**确定**。
Courier字体	命令。	执行cd /d C:/window命令，进入Windows系统文件夹。
斜体	表示参数、变量。	bae log list --instanceid Instance_ID
[]或者[a|b]	表示可选项，至多选择一个。	ipconfig [-all|-t]
{}或者{a|b}	表示必选项，至多选择一个。	switch {active|stand}

目录

1 开发准备

本文介绍在调用内容检测API前您需要完成的准备工作。

开通服务

- 前往阿里云官网注册一个阿里云账号。如果已注册账号，请跳过此步骤。
- 打开云盾内容安全产品试用页面，单击**立即开通**，正式开通服务。

创建访问密钥

在AccessKey管理页面管理您的AccessKeyID和AccessKeySecret。

说明：

AccessKey是您使用内容安全API时的密钥，拥有您的账号的完整权限。请您妥善保管它们，避免泄露，并定期更换您的AccessKey。

子账号支持

内容安全服务的内容检测API支持使用RAM子账号的AccessKeyID和AccessKeySecret。

您可以参照以下步骤，创建RAM子账号并完成授权：

1. 登录RAM控制台，创建子用户， 并选择生成AccessKeyID和AccessKeySecret。创建后请妥善保存该AccessKeyID和AccessKeySecret，用于后续调用API和SDK。

   具体操作，请参见#unique_4。
2. 完成子账号授权。向您创建的RAM子账号授予以下系统策略权限：AliyunYundunGreenWebFullAccess。

   具体操作，请参见#unique_5。

2 API概览

本文介绍内容检测API提供的所有配置管理接口，帮助您调用API配置自定义文本库、图库和管理OSS检测内容等。

导航

内容安全支持自定义文本库、图库和管理OSS检测内容，您可以在控制台执行相关操作，也可以通过接口调用进行操作，从而更好地融合到自有的管控系统链路中。

> **说明：**
> 强烈建议您优先选择引用我们的SDK进行快速接入，省去封装签名和请求协议的复杂操作。如果您的开发语言没有现成的SDK支持，您可以参见该API开发指南进行接入使用。关于SDK操作，请参见SDK概览。

内容检测API提供以下配置管理类接口。

类型	功能
自定义图库	通过API接口管理云盾控制台上的自定义图库。
自定义文本库	通过API接口管理云盾控制台上的自定义文本库。
OSS检测结果管理	通过API接口管理OSS违规检测的结果。

自定义相似图库

API	描述
DescribeImageLib	获取已创建的自定义图片库列表。
CreateImageLib	创建自定义图库。
UpdateImageLib	更新自定义图片库的名称及适用的**bizTypes**参数。
DeleteImageLib	删除自定义图片库。
DescribeImageFromLib	获取自定义图片库中已添加的图片列表。
DeleteImageFromLib	从自定义图片库中删除已添加的图片。
UploadImageToLib	向自定义库中添加图片元信息。

自定义文本库

API	描述
DescribeKeywordLib	获取已创建文本库列表。
CreateKeywordLib	创建自定义文本库。
UpdateKeywordLib	更新自定义文本库信息。
DeleteKeywordLib	删除自定义文本库。
CreateKeyword	向指定文本库中添加文本。
DescribeKeyword	搜索指定文本库中的文本。
DeleteKeyword	删除文本库中的文本。

OSS内容检测结果管理

API	描述
DescribeOssResultItems	获取OSS违规检测的检测结果数据。
MarkOssResult	对OSS违规检测结果进行审核和标记。
ExportOssResult	以文件的形式导出OSS违规检测结果。

3 构造请求

本文介绍调用内容检测API时构造调用请求的方法，适用于基于API URL发起HTTP、HTTPS请求的用户。

> **说明：**
> 我们提供多种编程语言的SDK及第三方SDK，可以免去您自己构造API请求的烦恼。更多信息，请参见SDK章节。

通信协议

支持HTTP、HTTPS协议。

请求方法

所有接口都支持POST、GET方法。

字符编码

所有接口的请求参数和返回结果编码方式都是UTF-8。

API请求结构

类型	描述	备注
API接入地址	具体API接口地址。	示例：http(https)://green.cn-shanghai.aliyuncs.com
公共参数	每个接口都包含有的通用参数。	具体请参见公共参数。
接口自定义参数	每个接口特有的参数。	详见每个API接口定义。

API接入地址

内容检测API配置管理服务（自定义图库、自定义文本库和OSS检测内容管理）的接入地址为：

```
http(s)://green.cn-shanghai.aliyuncs.com
```

> **注意：**
> 只支持以上地址。中国所有地域都请使用以上地址来调用服务，我们会自动同步到每个地域。

公共参数

名称	类型	是否必须	描述
Action	String	是	要执行的操作，即API接口名称。具体取值请参见API概览。
AccessKeyId	String	是	访问密钥ID。AccessKey用于调用 API。
Signature	String	是	您的签名。具体取值请参见签名机制。
SignatureMethod	String	是	签名方式。取值：HMAC-SHA1。
SignatureVersion	String	是	签名算法版本。取值：1.0。
SignatureNonce	String	是	签名唯一随机数。用于防止网络重放攻击，建议您每一次请求都使用不同的随机数。
Timestamp	String	是	请求的时间戳。按照ISO8601标准表示，并需要使用UTC时间，格式为yyyy-MM-ddTHH:mm:ssZ。示例：2018-01-01T12:00:00Z，表示北京时间2018年01月01日20点00分00秒。
Version	String	是	API的版本号，格式为YYYY-MM-DD。取值：2017-08-23。
Format	String	否	返回参数的语言类型。取值范围： • json • xml（默认）

接口自定义参数

请在API概览中，单击每个API进行查询。更多信息，请参见API概览。

签名机制

对于每一次HTTP或者HTTPS协议请求，我们会根据访问中的签名信息验证访问请求者身份。具体通过AccessKey ID和AccessKey Secret对称加密验证实现。

AccessKey相当于用户密码，AccessKey用于调用API，而用户密码用于登录阿里云控制台 。其中AccessKey ID是访问者身份，AccessKey Secret是加密签名字符串和服务器端验证签名字符串的密钥，必须严格保密谨防泄露。

1. 构造规范化请求字符串。

   a. 排序参数。排序规则以首字母顺序排序，排序参数包括公共参数和接口自定义参数，不包括公共请求参数中的**Signature**参数。

   > **说明：**
   > 当使用GET方法提交请求时，这些参数就是请求URL中的参数部分，即URL中?之后由&连接的部分。

   b. 编码参数。使用UTF-8字符集按照RFC3986规则编码请求参数和参数取值，编码规则如下：

   - 字符A~Z、a~z、0~9以及字符-、_、.、~不编码。
   - 其它字符编码成%XY的格式；其中XY是字符对应ASCII码的16进制。示例：半角双引号（"）对应%22。
   - 扩展的UTF-8字符，编码成%XY%ZA...的格式。
   - 空格（ ）编码成%20，而不是加号（+）。

     该编码方式与application/x-www-form-urlencodedMIME格式编码算法相似，但又有所不同。 如果您使用的是Java标准库中的java.net.URLEncoder，可以先用标准库中percentEncode编码，随后将编码后的字符中加号（+）替换为 %20、星号（*）替换为 %2A、%7E 替换为波浪号（~），即可得到上述规则描述的编码字符串。

     ```
     private static final String ENCODING = "UTF-8";
     private static String percentEncode(String value) throws UnsupportedEncodingE
     xception {
     return value != null ? URLEncoder.encode(value, ENCODING).replace("+", "%20").
     replace("*", "%2A").replace("%7E", "~") : null;
     }
     ```

   c. 使用等号（=）连接编码后的请求参数和参数取值。

   d. 使用与号（&）连接编码后的请求参数，注意参数排序与步骤 i 一致。

   现在，您得到了规范化请求字符串（CanonicalizedQueryString），其结构遵循API请求结构。

2. 构造签名字符串。

   a. 构造待签名字符串StringToSign。您可以同样使用percentEncode处理上一步构造的规范化请求字符串，规则如下：

   ```
   StringToSign=
    HTTPMethod + "&" + //HTTPMethod：发送请求的 HTTP 方法，例如 GET。
    percentEncode("/") + "&" + //percentEncode("/")：字符（/）UTF-8 编码得到的值，即
   %2F。
   ```

```
percentEncode(CanonicalizedQueryString) //您的规范化请求字符串。
```

b. 按照RFC2104的定义，计算待签名字符串StringToSign的HMAC-SHA1值。示例使用的是Java Base64编码方法。

```
Signature = Base64( HMAC-SHA1( AccessSecret, UTF-8-Encoding-Of(StringToSign ) ) )
```

> **说明：**
> 计算签名时，RFC2104规定的Key值是您的AccessKeySecret并加上与号（&），其ASCII值为38。

c. 添加根据RFC3986规则编码后的参数Signature到规范化请求字符串URL中。

代码示例

- 示例一：参数拼接法

  以调用**DescribeKeywordLib**查询词库为例。假设您获得了AccessKeyID=testid以及AccessKeySecret=testsecret，签名流程如下所示：

  1. 构造规范化请求字符串。

  ```
  AccessKeyId=testid&Action=DescribeKeywordLib&Format=XML&SignatureMethod=HMAC-SHA1&SignatureNonce=3ee8c1b8-83d3-44af-a94f-4e0ad82fd6cf&SignatureVersion=1.0&Timestamp=2016-02-23T12:46:24Z&Version=2014-05-26&ServiceModule=open_api
  ```

  2. 构造待签名字符串StringToSign。

  ```
  GET&%2F&AccessKeyId%3Dtestid%26Action%3DDescribeKeywordLib%26Format%3DXML%26SignatureMethod%3DHMAC-SHA1%26SignatureNonce%3D3ee8c1b8-83d3-44af-a94f-4e0ad82fd6cf%26SignatureVersion%3D1.0%26Timestamp%3D2016-02-23T12%253A46%253A24Z%26Version%3D2014-05-26%26ServiceModule%3Dopen_api
  ```

  3. 计算签名值。因为AccessKeySecret=testsecret，用于计算的Key为 testsecret&，计算得到的签名值为OLeaidS1JvxuMvnyHOwuJ+uX5qY=。示例使用的是Java Base64编码方法。

  ```
  Signature = Base64( HMAC-SHA1( AccessSecret, UTF-8-Encoding-Of(StringToSign ) ) )
  ```

  4. 添加RFC3986规则编码后的Signature=OLeaidS1JvxuMvnyHOwuJ%2BuX5qY%3D到步骤 i 的URL中。

  ```
  http://green.cn-shanghai.aliyuncs.com/?SignatureVersion=1.0&Action=DescribeKeywordLib&Format=XML&SignatureNonce=3ee8c1b8-83d3-44af-a94f-4e0ad82fd6cf&Version=2014-05-26&AccessKeyId=testid&Signature=OLeaidS1JvxuMvnyHOwuJ%
  ```

```
2BuX5qY%3D&SignatureMethod=HMAC-SHA1&Timestamp=2016-02-23T12%253A46
%253A24Z&ServiceModule=open_api
```

通过以上URL，您可以使用浏览器、curl或者wget等工具发起HTTP请求调用DescribeKeywordLib，查看自定义关键词库列表。

- 示例二：编程语言法

  依然以调用**DescribeKeywordLib**查询词库为例。假设您获得了AccessKeyID=testid以及AccessKeySecret=testsecret，并且假定所有请求参数放在一个JavaMap<String, String>对象里。

  1. 预定义编码方法。

  ```
  private static final String ENCODING = "UTF-8";
  private static String percentEncode(String value) throws UnsupportedEncodingE
  xception {
    return value != null ? URLEncoder.encode(value, ENCODING).replace("+", "%20").
  replace("*", "%2A").replace("%7E", "~") : null;
  }
  ```

  2. 预定义编码时间格式**Timestamp**。参数**Timestamp**必须符合ISO8601时间格式规范，并需要使用UTC时间，时区为+0。

  ```
  private static final String ISO8601_DATE_FORMAT = "yyyy-MM-dd'T'HH:mm:ss'Z'";
  private static String formatIso8601Date(Date date) {
    SimpleDateFormat df = new SimpleDateFormat(ISO8601_DATE_FORMAT);
    df.setTimeZone(new SimpleTimeZone(0, "GMT"));
    return df.format(date);
  }
  ```

  3. 构造请求字符串。

  ```
  final String HTTP_METHOD = "GET";
  Map parameters = new HashMap();
  // 输入请求参数。
  parameters.put("Action", "DescribeKeywordLib");
  parameters.put("Version", "2017-08-23");
  parameters.put("AccessKeyId", "testid");
  parameters.put("Timestamp", formatIso8601Date(new Date()));
  parameters.put("SignatureMethod", "HMAC-SHA1");
  parameters.put("SignatureVersion", "1.0");
  parameters.put("SignatureNonce", UUID.randomUUID().toString());
  parameters.put("Format", "XML");
  parameters.put("ServiceModule", "open_api");
  // 排序请求参数。
  String[] sortedKeys = parameters.keySet().toArray(new String[]{});
  Arrays.sort(sortedKeys);
  final String SEPARATOR = "&";
  // 构造stringToSign字符串。
  StringBuilder stringToSign = new StringBuilder();
  stringToSign.append(HTTP_METHOD).append(SEPARATOR);
  stringToSign.append(percentEncode("/")).append(SEPARATOR);
  StringBuilder canonicalizedQueryString = new StringBuilder();
  for(String key : sortedKeys) {
  // 注意编码key和value。
    canonicalizedQueryString.append("&")
    .append(percentEncode(key)).append("=")
  ```

```
  .append(percentEncode(parameters.get(key)));
}
// 注意编码canonicalizedQueryString。
stringToSign.append(percentEncode(
  canonicalizedQueryString.toString().substring(1)));
```

4. 签名。由于AccessKeySecret=testsecret，所以用于计算HMAC的Key为testsecret&，计算得到的签名值为OLeaidS1JvxuMvnyHOwuJ%2BuX5qY%3D。

```
// 以下是一段计算签名的示例代码。
final String ALGORITHM = "HmacSHA1";
final String ENCODING = "UTF-8";
key = "testsecret&";
Mac mac = Mac.getInstance(ALGORITHM);
mac.init(new SecretKeySpec(key.getBytes(ENCODING), ALGORITHM));
byte[] signData = mac.doFinal(stringToSign.getBytes(ENCODING));
String signature = new String(Base64.encodeBase64(signData));
```

增加签名参数后，按照RFC3986规则编码后的URL如下所示：

```
http://green.cn-shanghai.aliyuncs.com/?SignatureVersion=1.0&Action=DescribeKeywordLib&Format=XML&SignatureNonce=3ee8c1b8-83d3-44af-a94f-4e0ad82fd6cf&Version=2014-05-26&AccessKeyId=testid&Signature=OLeaidS1JvxuMvnyHOwuJ%2BuX5qY%3D&SignatureMethod=HMAC-SHA1&Timestamp=2016-02-23T12%253A46%253A24Z
```

5. 使用浏览器、curl或者wget等工具发送HTTP请求，获取返回结果。

API返回结果

请在API概览中单击每个API进行查询。更多信息，请参见API概览。

4 自定义图库

4.1 获取图库列表

调用DescribeImageLib接口获取自定义图库列表。

描述

业务接口：**DescribeImageLib**

获取自定义图库列表。

请求参数

名称	类型	是否必需	描述
ServiceModule	字符串	是	服务模块名称，取值： • open_api：用于内容检测API功能的图库 • website：用于站点检测功能的图库

返回参数

名称	类型	是否必需	描述
Id	数字型	是	主键ID。
ModifiedTime	字符串	是	最近一次修改时间。
Name	字符串	是	图库名称。
Code	字符串	是	图库编码。 **说明：** 在控制台中显示的图库编码，而在API接口中以主键ID标识图库。
ImageCount	字符串	是	图库中的图片数量。
Category	字符串	是	图库类型，取值： • BLACK：黑名单 • WHITE：白名单
Source	字符串	是	图库来源，取值： • MANUAL：手动创建 • FEEDBACK：根据用户反馈的结果自动创建

名称	类型	是否必需	描述
ServiceModule	字符串	是	服务模块名称，取值： • open_api：用于内容检测API功能的图库 • website：用于站点检测功能的图库
Scene	字符串	是	图库使用场景，取值： • PORN：智能鉴黄 • AD：广告识别 • ILLEGAL：暴恐涉政识别
BizTypes	列表	否	在**设置**页面的内容检测API的自定义图库位置显示的BizType。
Enable	布尔型	否	是否启用图库，true表示启用，false表示停用。

示例

请求示例

```
{
  "ServiceModule": "open_api"
}
```

返回示例

```
{
  "requestId": "6CF2815C-C8C7-4A01-B52E-FF6E24F53492",
  "data": {
    "TotalCount": 5,
    "ImageLibList": [
      {
        "Name": "测试a",
        "Source": "MANUAL",
        "ImageCount": 191231231230,
        "BizTypes": [
          1,
          2,
          3
        ],
        "Category": "BLACK",
        "ServiceModule": "open_api",
        "Scene": "PORN",
        "ModifiedTime": "2018-03-19 10:45:44 +0800",
        "Id": 1279,
        "Code": "808003055",
        "Enable": true
      },
      {
        "Name": "aa你好",
        "Source": "MANUAL",
        "ImageCount": 0,
        "BizTypes": [

        ],
```

```
            "Category": "BLACK",
            "ServiceModule": "open_api",
            "Scene": "PORN",
            "ModifiedTime": "2018-03-01 19:20:37 +0800",
            "Id": 1222,
            "Code": "808003043",
            "Enable": true
        }
    ]
  },
  "code": 200,
  "success": true
}
```


4.2 创建图库

调用CreateImageLib接口创建一个图库。

描述

业务接口：**CreateImageLib**

创建自定义相似图库。

说明：

图库的数量限制与通过控制台创建的图库共享。

请求参数

关于在请求中必须包含的公共请求参数，请参见公共参数。

请求body是一个结构体，结构说明如下：

名称	类型	是否必需	描述
Name	字符串	是	图库名称。
Category	字符串	是	图库类型，取值： • BLACK：黑名单 • WHITE：白名单
ServiceModule	字符串	是	服务模块名称，取值： • open_api：用于内容检测API功能。 • website：用于站点检测功能。
Scene	字符串	是	图库使用场景，取值： • PORN：智能鉴黄 • AD：广告识别 • ILLEGAL：暴恐涉政识别

名称	类型	是否必需	描述
BizTypes	列表	否	业务类型，在**设置**页面的内容检测API的自定义图库位置显示的BizType信息。 **说明：** 对于站点检测，该字段根据Instance生成。
Enable	布尔型	否	是否启用图库，true表示启用，false表示停用。

返回参数

返回结果说明，参见返回结果。

示例

请求示例

```
{
  "Name": "test1sdkdk",
  "BizTypes": [],
  "Category": "BLACK",
  "ServiceModule": "open_api",
  "ResourceType": "TEXT",
  "Scene": "PORN"
}
```

返回示例

```
{
  "requestId": "795D8871-4889-4C0F-A8B1-C7D2B990FF61",
  "code": 200
}
```

4.3 修改图库信息

调用UpdateImageLib接口修改指定图库的基本信息。

描述

业务接口：UpdateImageLib

修改图库的基本信息。

请求参数

关于在请求中必须包含的公共请求参数，请参考公共参数。

请求body是一个结构体，结构说明如下：

名称	类型	是否必需	描述
Id	数字型	是	主键ID。
Name	字符串	是	图库名称。
Category	字符串	是	图库类型，取值： • BLACK：黑名单 • WHITE：白名单
Scene	字符串	是	图库使用场景，取值： • PORN：智能鉴黄 • AD：广告识别 • ILLEGAL：暴恐涉政识别
BizTypes	列表	否	业务类型，在**设置**页面的内容检测API的自定义图库位置显示的BizType信息。 **说明：** 对于站点检测，该字段根据Instance生成。
Enable	布尔型	否	是否启用图库，true表示启用，false表示停用。

返回参数

返回结果说明，参见返回结果。

示例

请求示例

```
{
  "Id": 2147,
  "Name": "test2sdkdk",
  "BizTypes": []
}
```

返回示例

```
{
  "requestId": "795D8871-4889-4C0F-A8B1-C7D2B990FF61",
  "code": 200
```

```
}
```

4.4 删除图库

调用DeleteImageLib接口删除指定相似图库。

描述

业务接口：DeleteImageLib

删除指定相似图库。

> **说明：**
> 图库删除后无法恢复，请谨慎操作。

请求参数

关于在请求中必须包含的公共请求参数，请参考公共参数。

请求body是一个结构体，结构说明如下：

名称	类型	是否必需	描述
Id	数字型	是	图库主键ID。 **说明：** 请注意不要错误地传入图库Code字段值。

返回参数

返回结果说明，参见返回结果。

示例

请求示例

```
{
  "Id": 2147
}
```

返回示例

```
{
  "requestId": "795D8871-4889-4C0F-A8B1-C7D2B990FF61",
  "code": 200
```

```
}
```

4.5 查询图库中图片

调用DescribeImageFromLib接口查询指定图库中的图片。

描述

业务接口：**DescribeImageFromLib**

在指定图库中，以分页方式查询图片列表。

请求参数

关于在请求中必须包含的公共请求参数，请参见公共参数。

请求body是一个结构体，结构说明如下：

名称	类型	是否必需	描述
ImageLibId	数字型	是	图库主键ID。
TotalCount	数字型	否	返回图片数。
CurrentPage	字符串	否	返回的当前分页。
PageSize	数字串	是	返回的分页大小，即每个分页显示的图片数量。

返回参数

返回结果说明，参见返回结果。

名称	类型	是否必需	描述
TotalCount	数字型	否	返回图片数。
CurrentPage	字符串	否	返回的当前分页。
PageSize	数字串	否	返回的分页大小，即每个分页显示的图片数量。
ImageFromLibList	JSON数组	是	图片列表，具体结构描述见表 4-1: ImageFromLibList。

表 4-1: ImageFromLibList

名称	类型	是否必需	描述
Id	数字型	是	图片主键ID。
Image	字符串	是	图片URL地址。
Thumbnail	数字型	是	图片的缩略图URL地址。

示例

请求示例

```
{
  "ImageLibId": "2147"
}
```

返回示例


```
{
  "code": 200,
  "requestId": "73133169-C2FC-4A42-82FC-9C197BC5DCDE",
  "data": {
    "TotalCount": 12,
    "PageSize": 10,
    "CurrentPage": 1,
    "ImageFromLibList": [
      {
        "Id": 0,
        "Image": "https://gw.alicdn.com/bao/uploaded/TB1PDhjXrSYBuNjSspfXXcZCpXa-130-130.jpg",
        "Thumbnail": "https://gw.alicdn.com/bao/uploaded/TB1PDhjXrSYBuNjSspfXXcZCpXa-130-130.jpg"
      },
      {
        "Id": 1,
        "Image": "https://gw.alicdn.com/bao/uploaded/TB1ndhiXCCWBuNjy0FhXXb6EVXa-130-130.jpg",
        "Thumbnail": "https://gw.alicdn.com/bao/uploaded/TB1ndhiXCCWBuNjy0FhXXb6EVXa-130-130.jpg"
      },
      {
        "Id": 2,
        "Image": "https://gw.alicdn.com/bao/uploaded/TB1h0hiXCCWBuNjy0FhXXb6EVXa-130-130.jpg",
        "Thumbnail": "https://gw.alicdn.com/bao/uploaded/TB1h0hiXCCWBuNjy0FhXXb6EVXa-130-130.jpg"
      },
      {
        "Id": 3,
        "Image": "https://gw.alicdn.com/bao/uploaded/TB1PDhjXrSYBuNjSspfXXcZCpXa-130-130.jpg",
        "Thumbnail": "https://gw.alicdn.com/bao/uploaded/TB1PDhjXrSYBuNjSspfXXcZCpXa-130-130.jpg"
      },
      {
        "Id": 4,
        "Image": "https://gw.alicdn.com/bao/uploaded/TB1ndhiXCCWBuNjy0FhXXb6EVXa-130-130.jpg",
        "Thumbnail": "https://gw.alicdn.com/bao/uploaded/TB1ndhiXCCWBuNjy0FhXXb6EVXa-130-130.jpg"
      },
      {
        "Id": 5,
        "Image": "https://gw.alicdn.com/bao/uploaded/TB1h0hiXCCWBuNjy0FhXXb6EVXa-130-130.jpg",
        "Thumbnail": "https://gw.alicdn.com/bao/uploaded/TB1h0hiXCCWBuNjy0FhXXb6EVXa-130-130.jpg"
      },
      {
        "Id": 6,
```

```
      "Image": "https://gw.alicdn.com/bao/uploaded/TB1PDhjXrSYBuNjSspfXXcZCpXa-130-130.jpg",
      "Thumbnail": "https://gw.alicdn.com/bao/uploaded/TB1PDhjXrSYBuNjSspfXXcZCpXa-130-130.jpg"
    },
    {
      "Id": 7,
      "Image": "https://gw.alicdn.com/bao/uploaded/TB1ndhiXCCWBuNjy0FhXXb6EVXa-130-130.jpg",
      "Thumbnail": "https://gw.alicdn.com/bao/uploaded/TB1ndhiXCCWBuNjy0FhXXb6EVXa-130-130.jpg"
    },
    {
      "Id": 8,
      "Image": "https://gw.alicdn.com/bao/uploaded/TB1h0hiXCCWBuNjy0FhXXb6EVXa-130-130.jpg",
      "Thumbnail": "https://gw.alicdn.com/bao/uploaded/TB1h0hiXCCWBuNjy0FhXXb6EVXa-130-130.jpg"
    },
    {
      "Id": 9,
      "Image": "https://gw.alicdn.com/bao/uploaded/TB1PDhjXrSYBuNjSspfXXcZCpXa-130-130.jpg",
      "Thumbnail": "https://gw.alicdn.com/bao/uploaded/TB1PDhjXrSYBuNjSspfXXcZCpXa-130-130.jpg"
    },
    {
      "Id": 10,
      "Image": "https://gw.alicdn.com/bao/uploaded/TB1ndhiXCCWBuNjy0FhXXb6EVXa-130-130.jpg",
      "Thumbnail": "https://gw.alicdn.com/bao/uploaded/TB1ndhiXCCWBuNjy0FhXXb6EVXa-130-130.jpg"
    },
    {
      "Id": 11,
      "Image": "https://gw.alicdn.com/bao/uploaded/TB1h0hiXCCWBuNjy0FhXXb6EVXa-130-130.jpg",
      "Thumbnail": "https://gw.alicdn.com/bao/uploaded/TB1h0hiXCCWBuNjy0FhXXb6EVXa-130-130.jpg"
    },
    {
      "Id": 12,
      "Image": "https://gw.alicdn.com/bao/uploaded/TB1PDhjXrSYBuNjSspfXXcZCpXa-130-130.jpg",
      "Thumbnail": "https://gw.alicdn.com/bao/uploaded/TB1PDhjXrSYBuNjSspfXXcZCpXa-130-130.jpg"
    },
    {
      "Id": 13,
      "Image": "https://gw.alicdn.com/bao/uploaded/TB1ndhiXCCWBuNjy0FhXXb6EVXa-130-130.jpg",
      "Thumbnail": "https://gw.alicdn.com/bao/uploaded/TB1ndhiXCCWBuNjy0FhXXb6EVXa-130-130.jpg"
    },
    {
      "Id": 14,
      "Image": "https://gw.alicdn.com/bao/uploaded/TB1h0hiXCCWBuNjy0FhXXb6EVXa-130-130.jpg",
      "Thumbnail": "https://gw.alicdn.com/bao/uploaded/TB1h0hiXCCWBuNjy0FhXXb6EVXa-130-130.jpg"
    },
    {
      "Id": 15,
```

```
      "Image": "https://gw.alicdn.com/bao/uploaded/TB1PDhjXrSYBuNjSspfXXcZCpXa-130-130.jpg",
      "Thumbnail": "https://gw.alicdn.com/bao/uploaded/TB1PDhjXrSYBuNjSspfXXcZCpXa-130-130.jpg"
    },
    {
      "Id": 16,
      "Image": "https://gw.alicdn.com/bao/uploaded/TB1ndhiXCCWBuNjy0FhXXb6EVXa-130-130.jpg",
      "Thumbnail": "https://gw.alicdn.com/bao/uploaded/TB1ndhiXCCWBuNjy0FhXXb6EVXa-130-130.jpg"
    },
    {
      "Id": 17,
      "Image": "https://gw.alicdn.com/bao/uploaded/TB1h0hiXCCWBuNjy0FhXXb6EVXa-130-130.jpg",
      "Thumbnail": "https://gw.alicdn.com/bao/uploaded/TB1h0hiXCCWBuNjy0FhXXb6EVXa-130-130.jpg"
    },
    {
      "Id": 18,
      "Image": "https://gw.alicdn.com/bao/uploaded/TB1PDhjXrSYBuNjSspfXXcZCpXa-130-130.jpg",
      "Thumbnail": "https://gw.alicdn.com/bao/uploaded/TB1PDhjXrSYBuNjSspfXXcZCpXa-130-130.jpg"
    },
    {
      "Id": 19,
      "Image": "https://gw.alicdn.com/bao/uploaded/TB1ndhiXCCWBuNjy0FhXXb6EVXa-130-130.jpg",
      "Thumbnail": "https://gw.alicdn.com/bao/uploaded/TB1ndhiXCCWBuNjy0FhXXb6EVXa-130-130.jpg"
    },
    {
      "Id": 20,
      "Image": "https://gw.alicdn.com/bao/uploaded/TB1h0hiXCCWBuNjy0FhXXb6EVXa-130-130.jpg",
      "Thumbnail": "https://gw.alicdn.com/bao/uploaded/TB1h0hiXCCWBuNjy0FhXXb6EVXa-130-130.jpg"
    },
    {
      "Id": 21,
      "Image": "https://gw.alicdn.com/bao/uploaded/TB1PDhjXrSYBuNjSspfXXcZCpXa-130-130.jpg",
      "Thumbnail": "https://gw.alicdn.com/bao/uploaded/TB1PDhjXrSYBuNjSspfXXcZCpXa-130-130.jpg"
    },
    {
      "Id": 22,
      "Image": "https://gw.alicdn.com/bao/uploaded/TB1ndhiXCCWBuNjy0FhXXb6EVXa-130-130.jpg",
      "Thumbnail": "https://gw.alicdn.com/bao/uploaded/TB1ndhiXCCWBuNjy0FhXXb6EVXa-130-130.jpg"
    },
    {
      "Id": 23,
      "Image": "https://gw.alicdn.com/bao/uploaded/TB1h0hiXCCWBuNjy0FhXXb6EVXa-130-130.jpg",
      "Thumbnail": "https://gw.alicdn.com/bao/uploaded/TB1h0hiXCCWBuNjy0FhXXb6EVXa-130-130.jpg"
    }
  ]
}
```

```
}
```

4.6 删除图片

调用DeleteImageFromLib接口删除图库中的指定图片。

描述

业务接口：DeleteImageFromLib

删除图库中的指定图片，支持批量删除。

> **说明：**
> 图片删除后无法恢复，请谨慎操作。

请求参数

关于在请求中必须包含的公共请求参数，请参考公共参数。

请求body是一个结构体，结构说明如下：

名称	类型	是否必需	描述
Ids	列表型	是	图片ID列表。
ImageLibId	字符串	是	图库主键ID。

返回参数

返回结果说明，参见返回结果。

示例

请求示例

```
{
  "Id": [1,2],
  "ImageLibId": 2147
}
```

返回示例

```
{
  "requestId": "795D8871-4889-4C0F-A8B1-C7D2B990FF61",
  "code": 200
```

```
}
```

4.7 上传图片

调用UploadImageToLib接口上传图片。

描述

业务接口：**UploadImageToLib**

请求参数

名称	类型	是否必须	描述
ImageLibId	数字型	是	图库主键ID。
Urls	列表	是	图片列表，列表中的每一个元素是一张图片的OSS访问地址。

返回参数

名称	类型	描述
RequestId	字符串	本次请求的ID。

示例

请求示例

```
{
   "ImageLibId": "2147",
   "Urls": "[\"http://oss-example.com/1.jpg\"]"
}
```

返回示例

```
{
   "RequestId": "795D8871-4889-4C0F-A8B1-C7D2B990FF61",
   "code": 200
}
```

5 自定义文本库

5.1 获取文本库列表

本文提供了获取已添加的自定义文本库列表的具体接口及参数，旨在帮助您通过API接口自主管理您的文本库。您通过API接口管理的文本库可用于文本反垃圾检测，该方式与云盾内容控制台上的自定义文本库功能完全一致。

描述

业务接口：DescribeKeywordLib

获取自定义文本库列表。

请求参数

关于在请求中必须包含的公共请求参数，请参见公共参数。

请求body是一个结构体，结构说明如下：

名称	类型	是否必需	描述
ServiceModule	字符串	是	服务模块名称，取值： • open_api：用于内容检测API功能的文本库。 • website：用于站点检测功能的文本库。

返回参数

返回结果说明，参见返回结果。

返回body中的Data字段是JSON数组，每一个元素包含如下字段：

名称	类型	是否必需	描述
Id	数字型	是	主键ID。
ModifiedTime	字符串	是	最近一次修改时间。
Name	字符串	是	文本库名称。
Code	字符串	是	文本库编码。 **说明：** 在控制台中显示的文本库编码，而在API接口中以主键ID标识文本库。

名称	类型	是否必需	描述
Count	字符串	是	文本库中的样本数量。
Category	字符串	是	文本库类型，取值： • BLACK：黑名单 • WHITE：白名单 • REVIEW：灰名单
Source	字符串	是	文本库来源。 **说明：** 目前取值均为MANUAL，即手动方式添加。
ServiceModule	字符串	是	服务模块名称，取值： • open_api：用于内容检测API功能的文本库。 • website：用于站点检测功能的文本库。
BizTypes	列表	否	在**设置**页面的内容检测API的自定义文本库位置显示的BizType。
ResourceType	字符串	是	文本库类型，取值： • TEXT：文字文本库 • IMAGE：图片文本库 • VOICE：语言文本库
LibType	字符串	是	各应用场景中的文本库类型，取值： • **文本反垃圾场景** - textKeyword：关键词文本库 - similarText：相似文本库 • **图片广告场景** textKeyword：图片关键词文本库 • **语音反垃圾场景** voiceText：语音关键词文本库
MatchMode	字符串	否	匹配类型。取值： • fuzzy：模糊匹配 • precise：精确匹配
Enable	布尔型	否	是否启用文本库，true表示启用，false表示停用。

示例

请求示例

```
{
 "ServiceModule": open_api
}
```

返回示例

```
{
 "requestId": "795D8871-4889-4C0F-A8B1-C7D2B990FF61",
 "data": {
  "TotalCount": 11,
  "KeywordLibList": [
   {
    "Name": "test1sdkdk",
    "Source": "MANUAL",
    "BizTypes": [],
    "Category": "BLACK",
    "ServiceModule": "open_api",
    "Count": 3,
    "ResourceType": "TEXT",
    "Id": 2147,
    "ModifiedTime": "2018-09-04 14:17:56 +0800",
    "Code": "710001",
    "LibType": "textKeyword",
    "MatchMode":"fuzzy",
    "Enable":"true"
   },
   {
    "Name": "语音测试",
    "Source": "MANUAL",
    "BizTypes": [],
    "Category": "BLACK",
    "ServiceModule": "open_api",
    "Count": 3,
    "ResourceType": "VOICE",
    "Id": 2693,
    "ModifiedTime": "2018-09-03 17:11:04 +0800",
    "Code": "932001",
    "LibType": "voiceText",
    "MatchMode":"fuzzy",
    "Enable":"true"
   }
  ]
 },
 "code": 200,
 "success": true
```

```
}
```

5.2 创建文本库

本文提供了创建自定义文本库的具体接口及参数，旨在帮助您通过API接口自主管理您的文本库。您通过API接口管理的文本库可用于文本反垃圾检测，该方式与云盾内容控制台上的自定义文本库功能完全一致。

描述

业务接口：**CreateKeywordLib**

创建文本库。文本库适用于以下检测场景：

- 文本反垃圾：识别文本中包含的违规内容。
- 图文违规检测：识别图片中包含的广告和文字违规内容。
- 语音反垃圾：识别语音中包含的违规内容。

您还可以通过控制台操作创建文本库。更多信息，请参见创建和管理自定义文本库。

请求参数

关于在请求中必须包含的公共请求参数，请参见公共参数。

请求body是一个结构体，结构说明如下：

名称	类型	是否必须	描述
ServiceModule	字符串	是	服务模块名称，取值： • open_api：用于内容检测API功能的文本库。 • website：用于站点检测功能的文本库。
Name	字符串	是	文本库名称。
ResourceType	字符串	是	文本库应用的检测场景类型，取值： • TEXT：文本反垃圾 • IMAGE：图文违规检测 • VOICE：语音反垃圾
BizTypes	列表	否	在**设置**页面的内容检测API的自定义文本库位置显示的BizType。

名称	类型	是否必须	描述
Category	字符串	是	文本库类型，取值： • BLACK：黑名单 • WHITE：白名单 • REVIEW：灰名单
LibType	字符串	是	各应用场景中的文本库类型，取值： • 文本反垃圾 - textKeyword：关键词文本库 - similarText：相似文本库 • 图文违规识别 textKeyword：图片关键词文本库 • 语音反垃圾 voiceText：语音关键词文本库
MatchMode	字符串	否	匹配方式。取值： • fuzzy：模糊匹配 • precise：精确匹配
Enable	布尔型	否	是否启用文本库，true表示启用，false表示停用。

返回参数

返回结果说明，参见返回结果。

名称	类型	是否必须	描述
code	整型	是	错误码，和HTTP状态码一致（但有扩展）。 • 2xx：表示调用成功 • 4xx：表示请求有误 • 5xx：表示后端有误 具体请参见公共错误码。
msg	字符串	否	出现调用错误时，错误的描述信息。
requestId	字符串	是	唯一标识该请求的ID，可用于定位问题。
id	字符串	是	唯一标识一个文本库的id，可以用于后续对文本库中的内容进行操作。

示例

请求示例

```
{
  "Name": "test1sdkdk",
  "BizTypes": [],
  "Category": "BLACK",
  "ServiceModule": "open_api",
  "ResourceType": "TEXT",
  "LibType": "textKeyword"
}
```

返回示例

```
{
  "RequestId": "795D8871-4889-4C0F-A8B1-C7D2B990FF61",
  "code": 200,
  "Id":123
}
```

5.3 修改文本库

本文提供了修改自定义文本库基本信息的具体接口及参数，旨在帮助您通过API接口自主管理您的文本库。您通过API接口管理的文本库可用于文本反垃圾检测，该方式与云盾内容控制台上的自定义文本库功能完全一致。

描述

业务接口：**UpdateKeywordLib**

修改文本库的基本信息。

请求参数

关于在请求中必须包含的公共请求参数，请参见公共参数。

请求body是一个结构体，结构说明如下：

名称	类型	是否必需	描述
Id	数字型	是	文本库主键ID。
Name	字符串	是	文本库名称。
BizTypes	列表	否	在**设置**页面的内容检测API的自定义文本库位置显示的BizType。
Enable	布尔型	否	是否启用文本库，true表示启用，false表示停用。

返回参数

返回结果说明，参见返回结果。

示例

请求示例

```
{
  "Id": 2147,
  "Name": "test2sdkdk",
  "BizTypes": []
}
```

返回示例

```
{
  "requestId": "795D8871-4889-4C0F-A8B1-C7D2B990FF61",
  "code": 200
}
```

5.4 删除文本库

本文提供了删除自定义文本库的具体接口及参数，旨在帮助您通过API接口自主管理您的文本库。您通过API接口管理的文本库可用于文本反垃圾检测，该方式与云盾内容控制台上的自定义文本库功能完全一致。

描述

业务接口：**DeleteKeywordLib**

删除指定文本库。

说明:

文本库删除后无法恢复，请谨慎操作。

请求参数

关于在请求中必须包含的公共请求参数，请参见公共参数。

请求body是一个结构体，结构说明如下：

名称	类型	是否必需	描述
Id	数字型	是	文本库主键ID。

返回参数

返回结果说明，参见返回结果。

示例

请求示例

```
{
  "Id": 2147
}
```

返回示例

```
{
  "requestId": "795D8871-4889-4C0F-A8B1-C7D2B990FF61",
  "code": 200
}
```

5.5 添加文本

本文提供了在自定义文本库中添加关键词的具体接口及参数，旨在帮助您通过API接口自主管理您的文本库。您通过API接口管理的文本库可用于文本反垃圾检测，该方式与云盾内容控制台上的自定义文本库功能完全一致。

描述

业务接口：CreateKeyword

在指定文本库中添加关键词。

请求参数

关于在请求中必须包含的公共请求参数，请参见公共参数。

请求body是一个结构体，结构说明如下：

名称	类型	是否必需	描述
KeywordLibId	数字型	是	文本库主键ID。
Keywords	字符串	是	关键词列表。数组转成JSON字符串格式，例如 "[\"测试\"]"。

返回参数

返回结果说明，参见返回结果。

名称	类型	是否必需	描述
SuccessCount	数字型	是	已成功添加的关键词数量。
InvalidKeywordList	字符串	是	无效的关键词列表，即未添加成功的关键词。数组转成JSON字符串格式，例如 "[\"测试\"]"。

示例

请求示例

```
{
  "KeywordLibId": "2147",
  "Keywords": [test]
}
```

返回示例

```
{
  "code": 200,
  "data": {
    "InvalidKeywordList": [
      "test"
    ],
    "SuccessCount": 0
  }
}
```

5.6 搜索文本

本文提供了在自定义文本库中模糊搜索关键词的具体接口及参数，旨在帮助您通过API接口自主管理您的文本库。您通过API接口管理的文本库可用于文本反垃圾检测，该方式与云盾内容控制台上的自定义文本库功能完全一致。

描述

业务接口：DescribeKeyword

在指定文本库中以分页方式搜索关键词。

请求参数

关于在请求中必须包含的公共请求参数，请参见公共参数。

请求body是一个结构体，结构说明如下：

名称	类型	是否必需	描述
KeywordLibId	数字型	是	文本库主键ID。
Keyword	字符串	否	要搜索的关键词，支持模糊匹配。 **说明：** 如果不指定该参数，则返回该文本库中所有关键词。
TotalCount	数字型	否	返回关键词数。
CurrentPage	字符串	否	返回的当前分页。

名称	类型	是否必需	描述
PageSize	数字串	否	返回的分页大小，即每个分页显示的关键字数量。

返回参数

返回结果说明，参见返回结果。

名称	类型	是否必需	描述
TotalCount	数字型	否	返回关键词数。
CurrentPage	字符串	否	返回的当前分页。
PageSize	数字串	否	返回的分页大小，即每个分页显示的关键字数量。
KeywordList	JSON数组	是	关键词列表，具体结构描述见KeywordList。

表 5-1: KeywordList

名称	类型	是否必需	描述
Id	数字型	是	关键词主键ID。
CreateTime	字符串	是	创建时间。
Keyword	字符串	是	关键词。
HitCount	数字型	是	关键词的匹配命中次数。

示例

请求示例

```
{
 "KeywordLibId": "2147",
}
```

返回示例

```
{
 "code": 200,
 "requestId": "@guid",
 "data": {
  "TotalCount": 1000,
  "PageSize": 20,
  "CurrentPage": 1,
  "KeywordList": [
   {
    "Keyword": "的啊士大夫撒到",
    "HitCount": 101,
    "CreateTime": "2018-04-19 20:12:30 +0800",
    "Id": 1
   },
```

```
    {
      "Keyword": "习大大",
      "HitCount": 102,
      "CreateTime": "2018-04-19 20:12:30 +0800",
      "Id": 2
    }
  ]
 }
}
```


5.7 删除文本

本文提供了在自定义文本库中删除关键词的具体接口及参数，旨在帮助您通过API接口自主管理您的文本库。您通过API接口管理的文本库可用于文本反垃圾检测，该方式与云盾内容控制台上的自定义文本库功能完全一致。

描述

业务接口：DeleteKeyword

删除文本库中的关键词或相似文本。

> **说明：**
> 关键词删除后无法恢复，请谨慎操作。

请求参数

关于在请求中必须包含的公共请求参数，请参见公共参数。

请求body是一个结构体，结构说明如下：

名称	类型	是否必需	描述
Ids	列表型	否	要删除的关键词ID列表。 **说明：** **Ids**和**Keywords**必须至少传入一个。
Keywords	列表型	否	要删除的关键词列表。 **说明：** **Ids**和**Keywords**必须至少传入一个。
KeywordLibId	字符串	是	文本库主键ID。

返回参数

返回结果说明，参见返回结果。

示例

请求示例

```
{
  "Id": [1,2]
  "KeywordLibId": 2147
}
```

返回示例

```
{
  "requestId": "795D8871-4889-4C0F-A8B1-C7D2B990FF61",
  "code": 200
}
```

6 OSS检测内容管理

6.1 获取OSS检测结果

调用DescribeOssResultItems接口获取OSS违规检测的检测结果。

描述

业务接口：DescribeOssResultItems

分页显示OSS违规检测的检测结果。

请求参数

关于在请求中必须包含的公共请求参数，请参考公共参数。

请求body是一个结构体，结构说明如下：

名称	类型	是否必需	描述
PageSize	Number	否	每页显示的结果数量。
CurrentPage	Number	否	列表的页码。
StartDate	String	否	要查询的起始时间，格式为：yyyy-MM-dd HH:mm:ss Z。
EndDate	String	否	要查询的结束时间，格式为：yyyy-MM-dd HH:mm:ss Z。
Scene	String	是	要查询的检测场景，取值： • porn：涉黄 • terrorism：涉政
Suggestion	String	否（**Stock**为ture时必填）	要查询的结果建议类型。取值： • block：建议直接拦截 • review：建议人工审核 • pass：建议直接放行
Stock	Boolean	是	是否查询存量扫描结果。 • true表示查询存量扫描结果。 • false表示查询增量扫描结果。
MinScore	String	否	（仅适用于图片对象）要查询的结果的最低风险分值。

名称	类型	是否必需	描述
MaxScore	String	否	（仅适用于图片对象）要查询的结果的最高风险分值。
ResourceType	String	是	要查询的对象类型，取值： • IMAGE：图片对象 • VIDEO：视频对象
Bucket	String	否	要查询的OSS的Bucket名称。
QueryId	String	否	要查询的检测请求ID。

返回参数

返回结果说明，参见返回结果。

返回body中的Data字段是JSON数组，每一个元素包含如下字段：

名称	类型	是否必需	描述
TotalCount	Number	是	结果总数。
PageSize	Number	是	分页大小。
CurrentPage	Number	是	当前页码。
ScanResultList	List	是	扫描结果列表。
Id	String	否	数据ID。
Thumbnail	String	否	缩略图链接。 • 图片对象链接到图片缩略图。 • 视频对象链接到视频第一帧的缩略图。
Bucket	String	否	资源所在的OSS Bucket。
CreateTime	String	否	扫描时间。
Object	String	否	OSS object名称。
Score	Number	否	资源的风险分值，取值：0~100。分值越高，违规可能越大。
HandleStatus	Number	否	资源的处理状态，取值： • 0：未处理 • 1：删除 • 2：忽略 • 3：解冻 • 4：屏蔽 • 10：已处理

名称	类型	是否必需	描述
ResourceStatus	Number	否	资源的状态，取值： • 0：已删除 • 1：已冻结 • 2：可用
FrameResults	Object	否	（仅使用视频对象）存在风险的问题帧。
FrameResults.Thumbnail	String	否	该帧的缩略图链接。
FrameResults.Offset	String	否	该帧的位置，即距离视频开头的偏移量，单位为秒。
FrameResults.Url	String	否	该帧的链接。
NewUrl	String	否	原始文件链接地址。
Suggestion	String	否	扫描结果建议值。取值： • block：建议直接拦截 • review：建议人工审核 • pass：建议直接放行

示例

请求示例

```
{
 "Scene": "porn",
 "Suggestion": "block",
 "Stock": true,
 "ResourceType": "VIDEO"
}
```

返回示例

```
{
 "code": "200",
 "data": {
  "TotalCount": 2,
  "PageSize": 20,
  "CurrentPage": 1,
  "ScanResultList": [
   {
    "Id": 1235789,
    "TaskId": "222222",
    "DataId": "srffdffdfd2",
    "SequestTime": "1527410977000",
    "ScanFinishedTime": "1527410977000",
    "RequestTime": "1527410977000",
    "Suggestion": "review",
```

```
      "NewUrl": "https://gw.alicdn.com/bao/uploaded/LB1Q7lIoIbI8KJjy1zdXXbe1VXa.
mp4",
      "Thumbnail": "https://img.alicdn.com/tfs/TB1HPS9qb9YBuNjy0FgXXcxcXXa-200-200
.png",
      "CreateTime": "1527410977000",
      "Bucket": "uc-image",
      "Object": "post/image/eae1722368b6951ac05cfb68b1cdeb03.jpg",
      "Score": 90.9,
      "HandleStatus": 0,
      "ResourceStatus": 1,
      "FrameResults": [
       {
        "Thumbnail": "https://img.alicdn.com/tfs/TB1HPS9qb9YBuNjy0FgXXcxcXXa-200-
200.png",
        "Offset": 0,
        "Url": "https://img.alicdn.com/tfs/TB1HPS9qb9YBuNjy0FgXXcxcXXa-200-200.png"
       }
      ]
     },
     {
      "Id": 123579,
      "TaskId": "222222",
      "DataId": "srffdffdfd2",
      "SequestTime": "1527410977000",
      "ScanFinishedTime": "1527410977000",
      "RequestTime": "1527410977000",
      "Suggestion": "review",
      "NewUrl": "https://gw.alicdn.com/bao/uploaded/LB1Q7lIoIbI8KJjy1zdXXbe1VXa.
mp4",
      "Thumbnail": "https://img.alicdn.com/tfs/TB1HPS9qb9YBuNjy0FgXXcxcXXa-200-200
.png",
      "CreateTime": "1527410977000",
      "Bucket": "uc-image",
      "Object": "post/image/eae1722368b6951ac05cfb68b1cdeb03.jpg",
      "Score": 90.9,
      "HandleStatus": 0,
      "ResourceStatus": 1,
      "FrameResults": [
       {
        "Thumbnail": "https://img.alicdn.com/tfs/TB1HPS9qb9YBuNjy0FgXXcxcXXa-200-
200.png",
        "Offset": 0,
        "Url": "https://img.alicdn.com/tfs/TB1HPS9qb9YBuNjy0FgXXcxcXXa-200-200.png"
       }
      ]
     }
    ]
  },
  "requestId": "@guid",
  "successResponse": true
}
```


6.2 标记OSS检测结果

调用MarkOssResult接口对OSS检测结果进行审核标记。

描述

业务接口：**MarkOssResult**

根据OSS扫描结果的**数据ID**对OSS扫描结果进行人工审核标记，支持的操作包括：设置正常并忽略、设置违规并删除、设置正常并解冻。

> **说明：**
> 调用该接口前，请先通过获取OSS检测结果接口（DescribeOssResultItems）获取扫描结果数据中的ID。

请求参数

关于在请求中必须包含的公共请求参数，请参考公共参数。

请求body是一个结构体，结构说明如下：

名称	类型	是否必需	描述
Ids	List	是	要处理的OSS数据ID列表。 **说明：** 数据ID通过调用DescribeOssResultItems接口获得。
Operation	String	是	标记操作，取值： • ignore：忽略 • delete：删除 • unfreeze：解冻
Stock	String	是	是否是存量对象。 • true表示是存量对象。 • false表示是增量对象。
ResourceType	String	是	要标记的对象类型：取值： • IMAGE：图片 • VIDEO：视频
Scene	String	否	检测场景，取值： • porn：涉黄 • terrorism：涉政

返回参数

返回结果说明，参见返回结果。

示例

请求参数

```
{
  "Ids": ["123"],
  "Operation": "delete",
  "Stock": true,
  "ResouceType": "VIDEO"
}
```

返回参数

```
{
  "code": "200",
  "data": {
    "Code": 200,
    "Message": "OK"
  },
  "requestId": "@guid",
  "successResponse": true
}
```

6.3 导出OSS检测结果

调用ExportOssResult接口导出OSS检测结果。

描述

业务接口：ExportOssResult

以文件形式导出OSS违规检测的结果。

每次最多导出5,000条记录。导出的文件以txt形式存储，每行一条记录。每条记录包括以下内容：OSS扫描的Bucket名称，OSS扫描的文件（key），扫描的分数结果，扫描的处理建议（suggestion）。各个字段间以\t分割。

请求参数

关于在请求中必须包含的公共请求参数，请参考公共参数。

请求body是一个结构体，结构说明如下：

名称	类型	是否必需	描述
Stock	Boolean	是	是否导出存量结果。 • true表示导出存量结果。 • false表示导出增量结果。

名称	类型	是否必需	描述
ResourceType	String	否	要导出的结果的对象类型： • IMAGE：图片 • VEDIO：视频
Scene	String	否	要导出的结果的检测场景： • porn：涉黄 • terrorism：涉政
StartDate	String	否	要导出的结果的起始时间，格式为：yyyy-MM-dd HH:mm:ss Z。
EndDate	String	否	要导出的结果的结束时间，格式为：yyyy-MM-dd HH:mm:ss Z。
MinScore	String	否	（仅适用于图片对象）要导出的结果的最小风险分值。
MaxScore	String	否	（仅适用于图片对象）要导出的结果的最大风险分值。
Suggestion	String	否（**Stock**为true时必填）	要导出的对象的建议值，取值： • block：建议直接拦截 • review：建议人工审核 • pass：建议直接放行
Bucket	String	否	要导出的对象所在的OSS Bucket。
QueryId	String	否	要导出的对象的查询请求ID。
TotalCount	Number	否	结果总数。
PageSize	Number	否	分页大小。
CurrentPage	Number	否	当前页码。

返回参数

返回结果说明，参见返回结果。

返回body中的Data字段是JSON数组，每一个元素包含如下字段：

名称	类型	是否必需	描述
FileUrl	String	是	返回文件的链接地址。

示例

请求参数

```
{
  "Stock": true
}
```

返回参数

```
{
  "code": "200",
  "data": {
    "FileUrl": "https://img.alicdn.com/tfs/xxx.txt"
  },
  "requestId": "@guid",
  "successResponse": true
}
```