ALIBABA CLOUD

阿里云

安全加速 SCDN
开发指南

文档版本：20210825

阿里云

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. 阿里云网站上所有内容，包括但不限于著作、产品、图片、档案、资讯、资料、网站架构、网站画面的安排、网页设计，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
危险	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。	危险 重置操作将丢失用户配置数据。
警告	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。
注意	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。
说明	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`	命令或代码。	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.API参考 05
1.1. 简述 05
1.2. API调用方式 05
1.3. API概览 10
1.4. 附录 13

1.API参考

1.1. 简述

欢迎使用阿里云SCDN（安全加速）服务，您可以使用本文介绍的API对SCDN服务进行相关操作。

请确保在使用这些接口前，已充分了解SCDN产品说明、使用协议和收费方式。

- 当前SCDN OpenAPI版本号为：2017-11-15。
- 术语表见附录。

1.2. API调用方式

对SCDN API接口调用是通过向SCDN API的服务端地址发送HTTP GET请求，并按照接口说明在请求中加入相应请求参数来完成的；根据请求的处理情况，系统会返回处理结果。

请求结构

服务地址

SCDN API的服务接入地址为：`scdn.aliyuncs.com`。

通信协议

支持通过HTTP或HTTPS通道进行请求通信。为了获得更高的安全性，推荐您使用HTTPS通道发送请求。

请求方法

支持HTTP GET方法发送请求，这种方式下请求参数需要包含在请求的URL中。

请求参数

每个请求都需要指定要执行的操作，即Action参数（例如CreateSCDNServer），以及每个操作都需要包含的公共请求参数和指定操作所特有的请求参数。

字符编码

请求及返回结果都使用UTF-8字符集进行编码。

公共请求参数

公共请求参数是指每个接口都需要使用到的请求参数。

名称	类型	是否必须	描述
Format	String	否	返回值的类型，支持JSON与XML。默认为XML。
Version	String	是	API版本号，为日期形式：YYYY-MM-DD，本版本对应为2017-11-15。
AccessKeyId	String	是	阿里云颁发给用户的访问服务所用的密钥ID。

名称	类型	是否必须	描述
Signature	String	是	签名结果串，关于签名的计算方法，请参见签名机制。
SignatureMethod	String	是	签名方式，目前支持HMAC-SHA1。
Timestamp	String	是	请求的时间戳。日期格式按照ISO8601标准表示，并需要使用UTC时间。格式为：YYYY-MM-DDThh:mm:ssZ。例如，2017-11-15T12:00:00Z（为北京时间2017年11月15日20点0分0秒）。
SignatureVersion	String	是	签名算法版本，目前版本是1.0。
SignatureNonce	String	是	唯一随机数，用于防止网络重放攻击。用户在不同请求间要使用不同的随机数值。

请求示例：

```
https://scdn.aliyuncs.com/?Format=xml&Version=2013-01-10&Signature=Pc5WB8gokVn0xfeu%2FZV%2BiNM1dgI%3D&SignatureMethod=HMAC-SHA1&SignatureNonce=15215528852396&SignatureVersion=1.0&AccessKeyId=key-test&Timestamp=2012-06-01T12:00:00Z
```

公共返回参数

用户发送的每次接口调用请求，无论成功与否，系统都会返回一个唯一识别码RequestId给用户。

XML返回示例：

```
<?xml version="1.0" encoding="UTF-8"?>
<!--结果的根结点-->
<接口名称+Response>
    <!--返回请求标签-->
    <RequestId>4C467B38-3910-447D-87BC-AC049166F216</RequestId>
    <!--返回结果数据-->
</接口名称+Response>
```

JSON返回示例：

```
{
  "RequestId": "4C467B38-3910-447D-87BC-AC049166F216",
  /* 返回结果数据 */
}
```

返回结果

说明

- 调用API服务后返回数据采用统一格式，返回的HTTP状态码为2xx，代表调用成功。返回4xx或5xx的HTTP状态码代表调用失败。
- 调用成功返回的数据格式主要有XML和JSON两种，外部系统可以在请求时传入参数来制定返回的数据格式，默认为XML格式。
- 本文中的返回示例为了便于用户查看，做了格式化处理，实际返回结果没有进行换行、缩进等处理。

成功结果

XML返回示例：XML返回结果包括请求是否成功信息和具体的业务数据。

```
<?xml version="1.0" encoding="UTF-8"?>
<!--结果的根结点-->
<接口名称+Response>
  <!--返回请求标签-->
    <RequestId>4C467B38-3910-447D-87BC-AC049166F216</RequestId>
  <!--返回结果数据-->
</接口名称+Response>
```

JSON示例：

```
{
  "RequestId": "4C467B38-3910-447D-87BC-AC049166F216",
  /* 返回结果数据 */
}
```

错误结果

调用接口出错后，将不会返回结果数据。调用方可根据客户端错误和服务器端错误来定位错误原因。

当调用出错时，HTTP请求返回一个4xx或5xx的HTTP状态码。返回的消息体中是具体的错误代码及错误信息。另外还包含一个全局唯一的请求ID：RequestId和一个您该次请求访问的站点ID：HostId。在调用方找不到错误原因时，可以联系阿里云客服，并提供该HostId和RequestId，以便我们尽快帮您解决问题。

XML示例：

```
<?xml version="1.0" encoding="UTF-8"?>
<Error>
<RequestId>8906582E-6722-409A-A6C4-0E7863B733A5</RequestId>
  <HostId>scdn.aliyuncs.com</HostId>
  <Code>UnsupportedOperation</Code>
  <Message>The specified action is not supported.</Message>
</Error>
```

JSON示例：

```
{
  "RequestId": "8906582E-6722-409A-A6C4-0E7863B733A5",
  "HostId": "scdn.aliyuncs.com",
  "Code": "UnsupportedOperation",
  "Message": "The specified action is not supported."
}
```

签名机制说明

SCDN服务会对每个访问的请求进行身份验证，所以无论使用HTTP还是HTTPS协议提交请求，都需要在请求中包含签名（Signature）信息。SCDN通过使用 `Access Key ID` 和 `Access Key Secret` 进行对称加密的方法来验证请求的发送者身份。 `Access Key ID` 和 `Access Key Secret` 由阿里云官方颁发给访问者（可以通过阿里云官方网站申请和管理）。

> **说明**
>
> `Access Key ID` 用于标识访问者的身份， `Access Key Secret` 是用于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，只有阿里云和用户知道。

用户在访问时，按照下面的方法对请求进行签名处理：

1. 使用请求参数构造规范化的请求字符串（Canonicalized Query String）。
   i. 按照参数名称的字典顺序对请求中所有的请求参数（包括文档中描述的“公共请求参数”和给定了的请求接口的自定义参数，但不能包括“公共请求参数”中提到Signature参数本身）进行排序。

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

   ii. 对每个请求参数的名称和值进行编码。名称和值要使用 `UTF-8字符集` 进行URL编码，URL编码的编码规则如下：
      - 对于字符A~Z、a~z、0~9以及字符短划线（-）、下划线（_）、英文句号（.）、波浪线（~）不编码。
      - 对于其他字符编码成“%XY”的格式，其中XY是字符对应ASCII码的16进制表示。比如英文的双引号（”）对应的编码就是%22。
      - 对于扩展的UTF-8字符，编码成“%XY%ZA...”的格式。
      - 需要说明的是英文空格（ ）要被编码是%20，而不是加号（+）。

      一般支持URL编码的库（比如Java中的java.net.URLEncoder）都是按照“application/x-www-form-urlencoded”的MIME类型的规则进行编码的。实现时可以直接使用这类方式进行编码，把编码后的字符串中加号（+）替换成%20、星号（*）替换成%2A、%7E替换回波浪号（~），即可得到上述规则描述的编码字符串。

   iii. 对编码后的参数名称和值使用英文等号（=）进行连接。

iv. 再把英文等号连接得到的字符串按参数名称的字典顺序依次使用&符号连接，即得到规范化请求字符串。

2. 使用上一步构造的规范化字符串按照下面的规则构造用于计算签名的字符串：

```
StringToSign=
HTTPMethod + “&” +
percentEncode(“/”) + ”&” +
percentEncode(CanonicalizedQueryString)
```

其中HTTPMethod是提交请求用的HTTP方法，比 `GET` 。

percentEncode（“/”）是按照上面步骤1中b步骤描述的URL编码规则对字符“/”进行编码得到的值，即“%2F”。

percentEncode（CanonicalizedQueryString）是上面步骤1中a步骤对构造的规范化请求字符串，按步骤1中b步骤中描述的URL编码规则编码后得到的字符串。

3. 按照RFC2104的定义，使用上面的用于签名的字符串计算签名HMAC值。

> **注意**
>
> 计算签名时使用的Key就是用户持有的 `Access Key Secret` 并加上一个“&”字符（ASCII:38），使用的哈希算法是SHA1。

4. 按照Base64编码规则把上面的HMAC值编码成字符串，即得到签名值（Signature）。
5. 将得到的签名值作为Signature参数添加到请求参数中，即完成对请求签名的过程。

> **注意**
>
> 得到的签名值在作为最后的请求参数值提交给SCDN服务器的时候，要和其他参数一样，按照RFC3986的规则进行URL编码）。

以DescribeScdnService为例，签名前的请求URL为：

```
http://scdn.aliyuncs.com/?SignatureVersion=1.0&Format=JSON&Timestamp=2015-08-06T02:19:46Z&AccessKeyId=testid&SignatureMethod=HMAC-SHA1&Version=2014-11-11&Action=DescribeScdnService&SignatureNonce=9b7a44b0-3be1-11e5-8c73-08002700c460
```

那么StringToSign就是：

```
GET&%2F&AccessKeyId%3Dtestid&Action%3DDescribeScdnService&Format%3DJSON&SignatureMethod%3DHMAC-SHA1&SignatureNonce%3D9b7a44b0-3be1-11e5-8c73-08002700c460&SignatureVersion%3D1.0&Timestamp%3D2015-08-06T02%253A19%253A46Z&Version%3D2014-11-11
```

假如使用的 `Access Key Id` 是“testid”， `Access Key Secret` 是“testsecret”，用于计算HMAC的Key就是“testsecret&”，则计算得到的签名值是：

```
KkkQOf0ymKf4yVZLggy6kYiwgFs=
```

签名后的请求URL为（注意增加了Signature参数）：

```
http://scdn.aliyuncs.com/?SignatureVersion=1.0&Format=JSON&Timestamp=2015-08-06T02%3A19%3A46Z&AccessKeyId=testid&SignatureMethod=HMAC-SHA1&Version=2014-11-11&Signature=KkkQOf0ymKf4yVZLggy6kYiwgFs%3D&Action=DescribeScdnService&SignatureNonce=9b7a44b0-3be1-11e5-8c73-08002700c460
```

1.3. API概览

SCDN提供以下相关API接口。

服务操作接口

API	描述
DescribeScdnService	调用DescribeScdnService查询SCDN服务状态。包括：服务开通时间，频次攻击保护等数据。
CheckScdnService	调用CheckScdnService查询SCDN是否开通，是否欠费。
DescribeScdnUserQuota	调用DescribeScdnUserQuota查询用户资源上限及已使用情况。
OpenScdnService	调用OpenScdnService开通SCDN服务。只有开通后，才能进行域名操作。
DescribeScdnUserProtectInfo	调用DescribeScdnUserProtectInfo查询用户开通的服务信息。

域名操作接口

API	描述
DescribeScdnDomainDetail	调用DescribeScdnDomainDetail获取指定加速域名配置的基本信息。
DescribeScdnUserDomains	调用DescribeScdnUserDomains查询用户名下所有的安全加速域名。支持域名模糊匹配过滤和域名状态过滤。
AddScdnDomain	调用AddScdnDomain添加安全加速域名，一次只能提交一个加速域名。
UpdateScdnDomain	调用UpdateScdnDomain修改加速域名配置。
BatchDeleteScdnDomainConfigs	调用BatchDeleteScdnDomainConfigs批量删除域名配置。
DeleteScdnDomain	调用DeleteScdnDomain删除已添加的加速域名。
DescribeScdnDomainCertificateInfo	调用DescribeScdnDomainCertificateInfo获取指定加速域名证书信息。
DescribeScdnDomainCname	调用DescribeScdnDomainCname检测用户是否完成Cname，支持多个域名。

API	描述
StartScdnDomain	调用StartScdnDomain启用状态为“停用”的加速域名，将DomainStatus变更为online。
StopScdnDomain	调用StopScdnDomain停用某个加速域名，将DomainStatus变更为offline。

配置操作接口

API	描述
DescribeScdnDomainConfigs	调用DescribeScdnDomainConfigs查询域名配置，一次可查询多个功能配置。
DescribeScdnCcInfo	调用DescribeScdnCcInfo查询频次防护功能。
SetScdnCcInfo	调用SetScdnCcInfo设置频次防护功能。
DeleteScdnSpecificConfig	调用BatchDeleteScdnDomainConfigs删除加速域名的配置。
BatchSetScdnDomainConfigs	调用BatchSetScdnDomainConfigs批量配置域名。
DescribeScdnCertificateDetail	调用DescribeScdnCertificateDetail获取证书详细信息。
DescribeScdnCertificateList	调用DescribeScdnCertificateList获取证书列表信息。
SetScdnDomainCertificate	调用SetScdnDomainCertificate设置某域名下证书功能是否启用及修改证书信息。
SetScdnBotInfo	调用SetScdnBotInfo设置bot防护功能。
SetScdnDomainBizInfo	调用SetScdnDomainBizInfo设置域名业务类型。

刷新预热接口

API	描述
DescribeScdnRefreshQuota	调用DescribeScdnRefreshQuota查询刷新预热次数限制和余量。
RefreshScdnObjectCaches	调用RefreshScdnObjectCaches刷新节点上的文件内容。刷新指定URL内容至Cache节点，支持URL批量刷新。
PreloadScdnObjectCaches	调用PreloadScdnObjectCaches将源站的内容主动预热到L2 Cache节点上，用户首次访问可直接命中缓存，缓解源站压力。
DescribeScdnRefreshTasks	调用DescribeScdnRefreshTasks查询刷新、预热状态，是否在全网生效。

资源监控接口

API	描述
DescribeScdnCcQpsInfo	调用DescribeScdnCcQpsInfo查询频次控制监控数据
DescribeScdnCcTopIp	调用DescribeScdnCcTopIp查询频次攻击IP信息
DescribeScdnCcTopUrl	调用DescribeScdnCcTopUrl查询频次攻击URL信息
DescribeScdnDomainBpsData	调用DescribeScdnDomainBpsData获取安全加速域名的网络带宽监控数据。
DescribeScdnDomainTrafficData	调用DescribeScdnDomainTrafficData获取安全加速域名的网络流量监控数据，单位byte。
DescribeScdnDomainOriginBpsData	调用DescribeScdnDomainOriginBpsData获取安全加速域名的回源带宽监控数据。
DescribeScdnDomainOriginTrafficData	调用DescribeScdnDomainOriginTrafficData获取安全加速域名的回源流量监控数据，单位bit。
DescribeScdnDomainQpsData	调用DescribeScdnDomainQpsData获取安全加速域名的每秒访问次数QPS。
DescribeScdnDomainHitRateData	调用DescribeScdnDomainHitRateData获取安全加速域名的命中率，包括字节命中率和请求命中率。
DescribeScdnDomainHttpCodeData	调用DescribeScdnDomainHttpCodeData获取安全加速域名最小5分钟粒度的HTTP返回码占比数据。
DescribeScdnDomainIspData	调用DescribeScdnDomainIspData获取加速域名的天粒度的用户运营商分布数据的统计。
DescribeScdnDomainPvData	调用DescribeScdnDomainPvData获取加速域名最小1小时粒度的PV页面访问的统计数据。
DescribeScdnDomainRealTimeBpsData	调用DescribeScdnDomainRealTimeBpsData获取域名1分钟粒度带宽数据。
DescribeScdnDomainRealTimeByteHitRateData	调用DescribeScdnDomainRealTimeByteHitRateData获取域名1分钟粒度字节命中率的数据。
DescribeScdnDomainRealTimeHttpCodeData	调用DescribeScdnDomainRealTimeHttpCodeData获取加速域名1分钟粒度的HTTP返回码占比数据。
DescribeScdnDomainRealTimeQpsData	调用DescribeScdnDomainRealTimeQpsData获取域名1分钟粒度每秒访问次数的数据。
DescribeScdnDomainRealTimeReqHitRateData	调用DescribeScdnDomainRealTimeReqHitRateData获取域名1分钟粒度请求命中率的数据。
DescribeScdnDomainRealTimeSrcBpsData	调用DescribeScdnDomainRealTimeSrcBpsData获取域名一分钟粒度回源带宽的数据。

API	描述
DescribeScdnDomainRealTimeSrcTrafficData	调用DescribeScdnDomainRealTimeSrcTrafficData获取加速域名的1分钟回源流量监控数据，单位bit。
DescribeScdnDomainRealTimeTrafficData	调用DescribeScdnDomainRealTimeTrafficData获取加速域名的1分钟回源流量监控数据，单位bit。
DescribeScdnDomainRegionData	调用DescribeScdnDomainRegionData获取加速域名天粒度的用户区域分布的数据统计。
DescribeScdnDomainTopReferVisit	调用DescribeScdnDomainTopReferVisit获取加速域名某天的热门页面引用次数排名。
DescribeScdnDomainTopUrlVisit	调用DescribeScdnDomainTopUrlVisit获取加速域名某天内的热门URL列表。
DescribeScdnDomainUvData	调用DescribeScdnDomainUvData获取加速域名最小1小时粒度的UV页面独立访问的统计数据。
DescribeScdnTopDomainsByFlow	调用DescribeDcdnTopDomainsByFlow获取用户按流量排名的域名。

日志信息接口

API	描述
DescribeScdnDomainLog	调用DescribeScdnDomainLog获取指定域名原始访问日志的下载地址。

1.4. 附录

本文介绍了错误码和调用接口的说明示例。

错误代码表

客户端错误

ErrorCode错误代码	Message错误信息	HTTP状态码
OperationDenied	Your account does not open SCDN service yet.	403
InsufficientBalance	Your account does not have enough balance.	400
Forbidden.NotVerified	Your account is not verified yet.	403
UnsupportedOperation	The specified action is not supported.	400
NoSuchVersion	The specified version does not exist.	400

ErrorCode错误代码	Message错误信息	HTTP状态码
UnsupportedParameter	The parameter <parameter name> is not supported.	400
MissingParameter	The input parameter <parameter name> that is mandatory for processing this request is not supplied.	400
InvalidParameter	The specified parameter <parameter name> is not valid.Or The specified image does not support the specified instance type.	400
Throttling	Request was denied due to request throttling.	400
InvalidAccessKeyId.NotFound	The Access Key ID provided does not exist in our records.	404
Forbidden	User not authorized to operate on the specified resource.	403
Forbidden.RiskControl	This operation is forbidden by Aliyun Risk Control system.	403
Forbidden.AccessTooManyOthers Resource	This operator is forbidden because too many other one's resource to be accessed.	403
SignatureDoesNotMatch	The signature we calculated does not match the one you provided. Please refer to the API reference about authentication for details.	403
SignatureNonceUsed	The request signature nonce has been used.	400
IdempotentParameterMismatch	Request uses a client token in a previous request but is not identical to that request.	400
ChargeTypeViolation	Operations on this kind of resources are not permitted.	403
InsufficientBalance	Your account does not have enough balance.	400
QuotaExceeded	Living instances quota exceeded.	400
OperationDenied	Specified operation is denied as your resource is locked for security reasons.	403

ErrorCode错误代码	Message错误信息	HTTP状态码
RiskControl.Refused	Your action was.refused by RiskControl.	400
QuotaExceeded.Snapshot	Snapshot quota exceeded.	400
QuotaExceeded.Image	Image quota exceeded.	400

服务器端错误

错误代码	错误信息	HTTP状态码
InternalError	The request processing has failed due to some unknown error, exception or failure.	500
ServiceUnAvailable	The request has failed due to a temporary failure of the server.	503

如何调用接口

对SCDN服务接口的调用是通过向SCDN服务端发送HTTP请求（可以通过HTTP或HTTPS通道发送），并获取SCDN服务对该请求响应结果的过程。SCDN服务端在接收到用户请求后，对请求做必要的身份验证和参数验证，在所有验证成功后根据请求的指定参数提交或完成相应操作，并把处理的结果以HTTP响应地形式返回给调用者。

请求组成

请求由以下几个部分组成：

- HTTP方法——目前SCDN服务的所有接口只支持 `GET` 方法的调用。
- 请求URL——请求的服务地址、要执行的操作名称、操作参数和公共请求参数都包含在请求的URL中。
- 服务端地址：SCDN服务的域名是http://scdn.aliyuncs.com/和https://scdn.aliyuncs.com/。为了保证请求的安全性，强烈推荐您使用HTTPS通道。（HTTPS加入了SSL层对通信进行了加密，可以防止通信被截获而导致敏感信息泄露。）
- 操作名称：每个接口都需要指定要执行的操作名称，即Action参数。
- 操作参数：根据要执行的操作不同，需要传入不同的操作参数，详见每个接口的说明。
- 公共请求参数：包含时间戳、签名信息等每个请求都要包含的参数。

为了使服务端能够正确地验证用户的身份并授权请求执行，请求在提交前要进行签名处理。签名的规则参见签名机制一节。

在服务端对请求处理完成后，会返回响应结果。响应结果分为成功结果和错误消息，格式描述参见JSON返回示例：

```
{
  "RequestId": "4C467B38-3910-447D-87BC-AC049166F216",
  /* 返回结果数据 */
}
```

返回结果：客户端可以解析响应的消息体，得到执行结果。

调用示例

以DescribeScdnService接口为例：

对应的Action是DescribeScdnService。在添加了所有公共请求参数（除Signature）后，请求的URL是（为了便于阅读，这里是进行URL编码前的URL）：

```
http://scdn.aliyuncs.com/?TimeStamp=2012-12-26T10:33:56Z&Format=XML&AccessKeyId=testid&Action= DescribeScdnService &SignatureMethod=HMAC-SHA1&SignatureNonce=NwDAxvLU6tFE0DVb&Version=2013-01-10&SignatureVersion=1.0
```

按照签名计算规则，先构造出规范化请求字符串（Canonicalized Query String），如下：

```
http://scdn.aliyuncs.com/?TimeStamp=2012-12-26T10:33:56Z&Format=XML&AccessKeyId=testid&Action= DescribeScdnService &SignatureMethod=HMAC-SHA1&SignatureNonce=NwDAxvLU6tFE0DVb&Version=2013-01-10&SignatureVersion=1.0
```

再构造出用于签名的字符串StringToSign值为：

```
GET&%2F&AccessKeyId%3Dtestid&Action% DescribeScdnService &Format%3DXML&SignatureMethod%3DHMAC-SHA1&SignatureNonce%3DNwDAxvLU6tFE0DVb&SignatureVersion%3D1.0&TimeStamp%3D2012-12-26T10%253A33%253A56Z&Version%3D2013-01-10
```

以下Java示例代码演示了如何添加公共请求参数、如何构造用请求参数构造规范化请求字符串，以及如何构造StringToSign字符串。示例假定所有请求参数放在一个Map <String, String>对象里，使用的Access Key ID是“testid”。

```
final String HTTP_METHOD = "GET";
.........................................
```

其中需要注意的是，TimeStamp参数要求符合ISO8601规范，并注意使用UTC时间，否则会遇到错误。下面的示例代码演示了如何生成符合规范的TimeStamp字符串：

```
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);
  }
```

生成规范化请求字符串（示例中的canonicalizedQueryString变量），以及stringToSign时，都需要进行必要的编码。编码的规则在签名机制一节中有详细描述。下面的示例代码演示了如何用java.net.URLEncoder类完成编码：

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

假定使用的 `Access Key Id` 是“testid”， `Access Key Secret` 是“testsecret”，用于计算HMAC的Key就是“testsecret&”，最终计算得到的签名值为：

```
SDFQNvyH5rtkc9T5Fwo8DOjw5hc=
```

计算签名的示例代码（Java）：

```
// 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，请按照RFC3986规则进行URL编码后得到的

```
http://scdn.aliyuncs.com/?TimeStamp=2012-12-26T10%3A33%3A56Z&Format=XML&AccessKeyId=testid&Action= DescribeScdnService &SignatureMethod=HMAC-SHA1&RegionId=region1&SignatureNonce=NwDAxvLU6tFE0DVb&Version=2012-09-13&SignatureVersion=1.0&Signature=SDFQNvyH5rtkc9T5Fwo8DOjw5hc%3d
```

接下来，通过HTTP请求的方式向上面的URL地址发送HTTP请求，并得到SCDN服务端的响应结果（示例）：

```
none
```

通过解析这个XML结果即可以得到所有可用的地域ID和LocalName的列表。如果在提交请求时，指定Format参数为JSON，那么返回结果的格式为JSON格式。

如何保证幂等性

当通过调用创建实例接口在SCDN中创建云服务器时，如果遇到了请求超时或服务器内部错误时，客户端可能会尝试重发请求，这时客户端可以通过提供可选参数 `ClientToken` 避免服务器创建出比预期要多的实例，也就是通过提供 `ClientToken` 参数保证请求的幂等性。 `ClientToken` 是一个由客户端生成的唯一的、大小写敏感、不超过64个ASCII字符的字符串。

如果用户使用同一个 `ClientToken` 值调用创建实例接口，则服务端会返回相同的请求结果，包含相同的InstanceId。因此用户在遇到错误进行重试的时候，可以通过提供相同的ClientToken值，来确保SCDN只创建一个实例，并得到这个实例的InstanceId。

如果用户提供了一个已经使用过的ClientToken，但其他请求参数不同，则SCDN会返回*IdempotentParameterMismatch*的错误代码。但需要注意的是，SignatureNonce、Timestamp和Signature参数在重试时是需要变化的，因为scdn使用SignatureNonce来防止重放攻击，使用Timestamp来标记每次请求时间，所以再次请求必须提供不同的SignatureNonce和Timestamp参数值，这同时也会导致Signature值的变化。

通常客户端只需要在500（InternetError）或503（ServiceUnAvailable）错误、或者无法得到响应结果的情况下进行重试操作。返回结果是200时，重试可以得到上次相同的结果，但不会对服务端状态带来任何影响。而对4xx的返回错误，除非提示信息里明确出现“try it later”，通常重试也是不能成功的。

欠费状态下的API行为

下列表中“-”表示无关，“正常逻辑”表示按照接口的正常逻辑执行并返回结果。

账户欠费时

接口	行为
OpenScdnService	正常