

# 阿里云 轻量级分布式应用服务

API参考

文档版本：20190905

## 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
	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。	 <b>禁止：</b> 重置操作将丢失用户配置数据。
	该类警示信息可能导致系统重大变更甚至故障，或者导致人身伤害等结果。	 <b>警告：</b> 重启操作将导致业务中断，恢复业务所需时间约10分钟。
	用于补充说明、最佳实践、窍门等，不是用户必须了解的内容。	 <b>说明：</b> 您也可以通过按Ctrl + A选中全部文件。
>	多级菜单递进。	设置 > 网络 > 设置网络类型
<b>粗体</b>	表示按键、菜单、页面名称等UI元素。	单击 <b>确定</b> 。
<code>courier</code> 字体	命令。	执行 <code>cd /d C:/windows</code> 命令，进入Windows系统文件夹。
<code>##</code>	表示参数、变量。	<code>bae log list --instanceid</code> <code>Instance_ID</code>
<code>[ ]</code> 或者 <code>[a b]</code>	表示可选项，至多选择一个。	<code>ipconfig [-all -t]</code>
<code>{ }</code> 或者 <code>{a b}</code>	表示必选项，至多选择一个。	<code>swich {stand   slave}</code>

# 目录

---

法律声明.....	I
通用约定.....	I
1 API概览.....	1
2 安装新版 Java SDK.....	3
3 调用方式.....	6
4 签名机制.....	7
5 公共请求和返回头.....	11
6 命名空间和VPC.....	13
6.1 CreateNamespace.....	13
6.2 DeleteNamespace.....	16
6.3 DescribeNamespace.....	18
6.4 DescribeNamespaces.....	20
6.5 UpdateNamespace.....	23
6.6 DescribeNamespaceList.....	25
7 应用信息.....	29
7.1 DescribeApplicationConfig.....	29
7.2 DescribeRegions.....	33
7.3 DescribeInstanceLog.....	35
7.4 DescribeComponents.....	37
7.5 DescribeEdasContainers.....	39
7.6 DescribeApplicationImage.....	41
7.7 DescribeApplicationInstances.....	44
7.8 DescribeApplicationGroups.....	47
7.9 ListApplications.....	50
7.10 QueryResourceStatics.....	52
7.11 ListLogConfigs.....	54
8 应用生命周期.....	58
8.1 DescribeApplicationStatus.....	58
8.2 DeployApplication.....	61
8.3 CreateApplication.....	67
8.4 DeleteApplication.....	74
8.5 StopApplication.....	76
8.6 RescaleApplicationVertically.....	79
8.7 StartApplication.....	82
8.8 ConfirmPipelineBatch.....	84
8.9 ListChangeOrders.....	86
8.10 AbortAndRollbackChangeOrder.....	90
8.11 DescribeChangeOrder.....	93

8.12 DescribeInstanceSpecifications.....	97
8.13 RescaleApplication.....	99
8.14 RestartApplication.....	102
8.15 AbortChangeOrder.....	104
<b>9 SLB.....</b>	<b>107</b>
9.1 BindSlb.....	107
9.2 UnbindSlb.....	110
9.3 DescribeApplicationSlbs.....	112
<b>10 微服务列表.....</b>	<b>116</b>
10.1 ListConsumedServices.....	116
10.2 ListPublishedServices.....	118
<b>11 公共错误码.....</b>	<b>121</b>



# 1 API概览

轻量级分布式应用服务提供以下相关API接口。

## 命名空间和VPC

API	描述
<a href="#">CreateNamespace</a>	创建命名空间
<a href="#">DeleteNamespace</a>	删除命名空间
<a href="#">DescribeNamespace</a>	查询命名空间详细信息
<a href="#">DescribeNamespaces</a>	查询命名空间列表
<a href="#">UpdateNamespace</a>	更新命名空间信息
<a href="#">DescribeNamespaceList</a>	获取命名空间列表

## 应用信息

API	描述
<a href="#">DescribeApplicationConfig</a>	获取应用配置信息。
<a href="#">DescribeRegions</a>	使用 DescribeRegions 查询可用地域。
<a href="#">DescribeInstanceLog</a>	获取实例日志
<a href="#">DescribeComponents</a>	获取应用创建部署时所需的组件版本
<a href="#">DescribeEdasContainers</a>	获取应用微服务容器组件列表
<a href="#">DescribeApplicationImage</a>	描述应用镜像信息
<a href="#">DescribeApplicationInstances</a>	获取应用实例列表
<a href="#">DescribeApplicationGroups</a>	获取应用实例分组
<a href="#">ListApplications</a>	获取应用列表
<a href="#">QueryResourceStatics</a>	获取应用的资源使用量
<a href="#">ListLogConfigs</a>	获取应用日志列表

## 应用生命周期

API	描述
<a href="#">DescribeApplicationStatus</a>	获取应用的状态信息
<a href="#">DeployApplication</a>	部署应用
<a href="#">CreateApplication</a>	创建一个SAE应用

API	描述
<a href="#">DeleteApplication</a>	删除应用
<a href="#">StopApplication</a>	停止应用
<a href="#">RescaleApplicationVertically</a>	改变应用实例规格
<a href="#">StartApplication</a>	启动应用
<a href="#">ConfirmPipelineBatch</a>	是否开始一下批次
<a href="#">ListChangeOrders</a>	获取变更单列表
<a href="#">AbortAndRollbackChangeOrder</a>	中止或回滚变更单
<a href="#">DescribeChangeOrder</a>	查询变更单信息
<a href="#">DescribeInstanceSpecification</a>	获取应用实例规格信息
<a href="#">RescaleApplication</a>	应用扩缩
<a href="#">RestartApplication</a>	重启应用
<a href="#">AbortChangeOrder</a>	中止变更单

## SLB

API	描述
<a href="#">BindSlb</a>	为应用绑定SLB
<a href="#">UnbindSlb</a>	解绑内网或公网SLB
<a href="#">DescribeApplicationSlbs</a>	获取应用SLB配置信息

## 微服务列表

API	描述
<a href="#">ListConsumedServices</a>	获取订阅的微服务列表
<a href="#">ListPublishedServices</a>	获取发布的微服务列表



## 2 安装新版 Java SDK

相较于旧版 Java SDK 直接开放下载，阿里云新版 Java SDK 采用了 Maven 的方式分发。为方便统一管理，阿里云所有的 Java SDK 均统一在同一个 Maven 项目中。

### 下载 Maven 软件

访问 [Maven 官方下载页面](#) 下载对应操作系统的 Maven 软件。Checksum 文件可供您校验下载文件是否正确无误。以下截图来自于 Windows 7 64 位操作系统，使用环境是 Eclipse Luna。

Files	Link	Checksums	Signature
Binary tar.gz archive	<a href="#">apache-maven-3.6.1-bin.tar.gz</a>	<a href="#">apache-maven-3.6.1-bin.tar.gz.sha512</a>	<a href="#">apache-maven-3.6.1-bin.tar.gz.asc</a>
Binary zip archive	<a href="#">apache-maven-3.6.1-bin.zip</a>	<a href="#">apache-maven-3.6.1-bin.zip.sha512</a>	<a href="#">apache-maven-3.6.1-bin.zip.asc</a>
Source tar.gz archive	<a href="#">apache-maven-3.6.1-src.tar.gz</a>	<a href="#">apache-maven-3.6.1-src.tar.gz.sha512</a>	<a href="#">apache-maven-3.6.1-src.tar.gz.asc</a>
Source zip archive	<a href="#">apache-maven-3.6.1-src.zip</a>	<a href="#">apache-maven-3.6.1-src.zip.sha512</a>	<a href="#">apache-maven-3.6.1-src.zip.asc</a>

### 安装 Java SDK

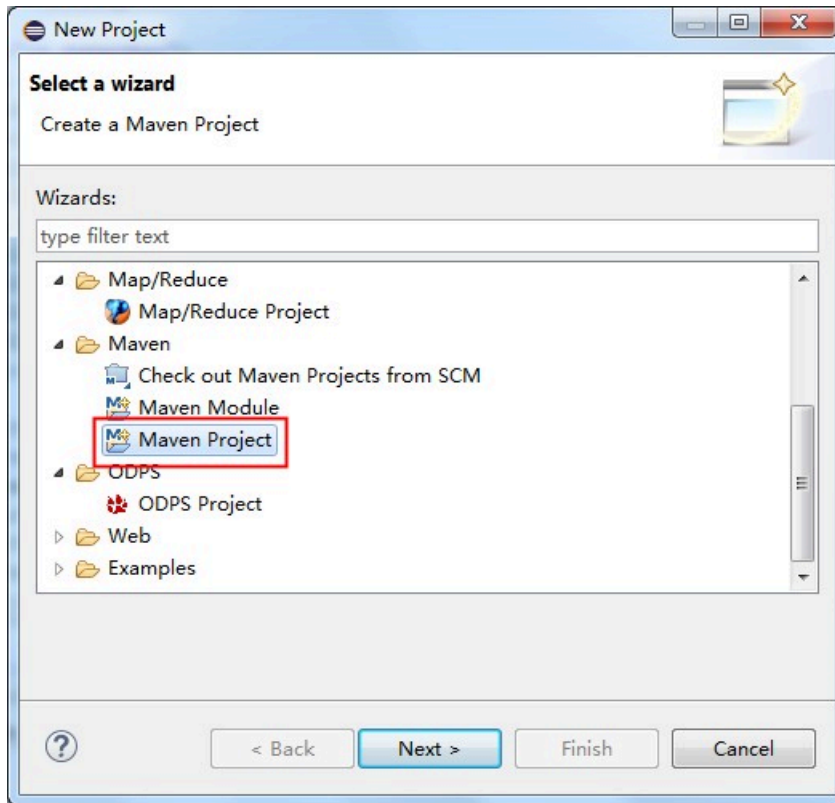
1. 打开 [Java SDK 下载页面](#)，将阿里云的 SDK 存放的 Maven 库加入到 Maven 软件中。
2. 找到之前下载的 Maven 压缩包并解压，将下列 Maven 库信息添加到 conf 文件夹下的 settings.xml 文件中。

```
<repositories>
  <repository>
    <id>sonatype-nexus-staging</id>
    <name>Sonatype Nexus Staging</name>
    <url>https://oss.sonatype.org/service/local/staging/deploy/maven2/</url>
    <releases>
      <enabled>true</enabled>
    </releases>
    <snapshots>
      <enabled>true</enabled>
    </snapshots>
  </repository>
```

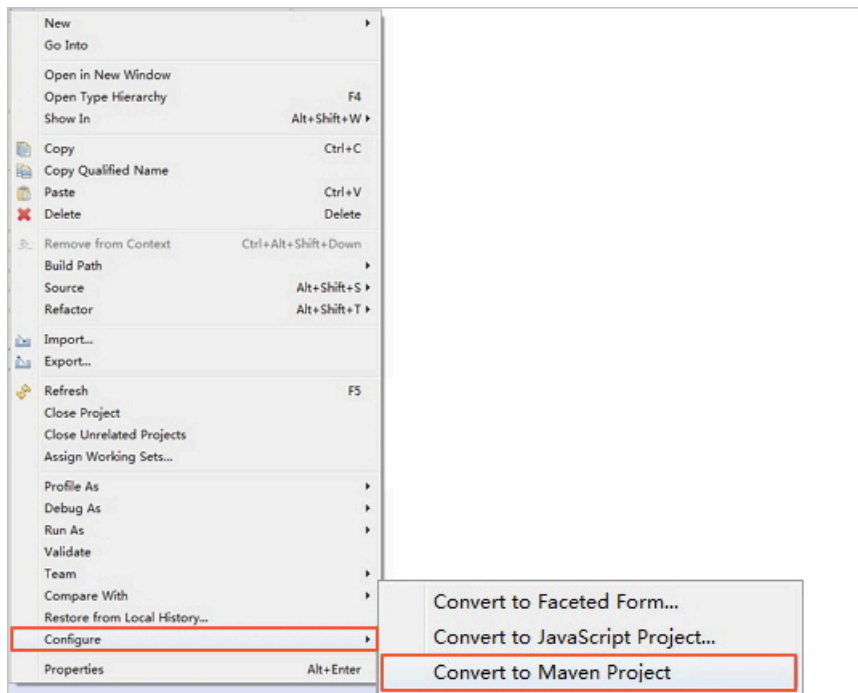
```
</repositories>
```

### 3. 通过以下任一方式创建 Maven 项目：

- 方式一：在 Eclipse 添加一个 Maven 项目。



- 方式二：将已有的项目转换为 Maven 项目。



- 打开 Maven 项目下的 `pom.xml` 文件，将 `dependency` 加入 Maven 项目中。您可以根据自身需要添加不同的依赖。



说明:

`aliyun-java-sdk-core`需设置为4.3.2版本。

```
<dependencies>
  <dependency>
    <groupId>com.aliyun</groupId>
    <artifactId>aliyun-java-sdk-core</artifactId>
    <version>4.3.2</version>
  </dependency>
  <dependency>
    <groupId>com.aliyun</groupId>
    <artifactId>aliyun-java-sdk-sae</artifactId>
    <version>1.0.0</version>
  </dependency>
  <dependency>
    <groupId>com.aliyun</groupId>
    <artifactId>aliyun-java-sdk-yundun</artifactId>
    <version>2.1.3</version>
  </dependency>
  <dependency>
    <groupId>com.aliyun</groupId>
    <artifactId>aliyun-java-sdk-slb</artifactId>
    <version>2.0.0-rc1</version>
  </dependency>
</dependencies>
```

- 保存更改。Maven Dependencies 中会自动下载并加入 `.jar` 后缀的相应阿里云 SDK。

#### 区分旧版 SDK 与新版 SDK

若您使用的是旧版 SDK，建议您切换为新版 SDK，获取最新功能。通过下表中的 SDK 参数，可以识别 SDK 版本。

对比项	旧版 SDK	新版 SDK
如何提交请求	<code>execute()</code>	<code>getAcResponse()</code>
如何存放 <code>AccessKey</code> 和 <code>AccessKeySecret</code> 的类	<code>AliyunClient</code>	<code>IClientProfile</code>
如何存放凭据对象	<code>new DefaultAliyunClient(APIUrl, Access Key, Access Key Secret)</code>	<code>DefaultProfile.getProfile(RegionId, Access Key, Access Key Secret)</code>
包名前缀	<code>com.aliyun.api</code>	<code>com.aliyuncs</code>

## 3 调用方式

轻量级分布式应用服务API接口的调用是向轻量级分布式应用服务的服务端地址发送HTTP请求，并按照接口说明在请求中加入相应请求参数，调用后系统会返回处理结果。请求及返回结果都使用 UTF-8 字符集进行编码。

### 服务地址

轻量级分布式应用服务的API接入地址为 [sae.cn-hangzhou.aliyuncs.com](http://sae.cn-hangzhou.aliyuncs.com)。

### 通信协议

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

### 请求语法

轻量级分布式应用服务提供RESTful API供您使用，其请求结构如下：

```
HTTPMethod /resource_URI_parameters
RequestHeader
RequestBody
```

#### 其中：

- HTTPMethod：请求使用的方法，包括PUT、POST、GET、DELETE。
- resource\_URI\_parameters：请求要调用的资源标示符，如/`cluster`。
- RequestHeader：请求头信息，通常包含API的版本、Host、Authorization等信息。更多信息，请参见[ZH-CN\\_TP\\_1068757\\_V1.dita#concept\\_1340341](#)。

请求参数由公共请求头和API自定义参数组成。公共请求头中包含API版本号、身份验证等信息。

- RequestBody：请求参数。

#### 示例：

```
GET /pop/v1/sam/app/describeApplicationConfig HTTP/1.1
公共请求头
```

### 请求编码

请求及返回结果都使用UTF-8字符集进行编码。

## 4 签名机制

为保证API的安全调用，在调用API时阿里云会对每个API请求通过签名（Signature）进行身份验证。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信息。

### 概述

RESTful API需要按如下格式在API请求头（request header）中添加Authorization参数进行签名：

```
Authorization:acs:AccessKeyId:Signature
```

其中：

- acs：Alibaba Cloud Service的缩写，固定标识不可修改。
- AccessKeyId：调用者调用API所用的密钥ID。
- Signature：使用AccessKey Secret对请求进行对称加密的签名。

### 计算签名

签名算法遵循RFC 2104 HMAC-SHA1规范，使用AccessSecret对编码、排序后的整个请求串计算HMAC值作为签名。签名的元素是请求自身的一些参数，由于每个API请求内容不同，所以签名的结果也不尽相同。

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

完成以下操作，计算签名：

#### 1. 构建待签名字符串。

待签名字符串（StringToSign）是API请求拼装的字符串，用于计算签名，包括：

- HTTP协议Header
- 阿里云协议Header (CanonicalizedHeaders)
- 规范资源 (CanonicalizedResource)
- Body

待签名字符串必须按照以下顺序构造：

```
StringToSign =  
  //http协议Header  
  HTTP-Verb + "\n" +  
  Accept + "\n" +  
  Content-MD5 + "\n" + //Body的MD5值放在此处  
  Content-Type + "\n" +
```

```
Date + "\n" +
//阿里云协议header(CanonicalizedHeaders)
CanonicalizedHeaders +
//签名中如何包含CanonicalizedResource (规范资源)
CanonicalizedResource
```

#### 示例：原始请求

```
POST /stacks?name=test_alert&status=COMPLETE HTTP/1.1
Host: ***.aliyuncs.com
Accept: application/json
Content-MD5: ChDfdfwC+Tn874znq7Dw7Q==
Content-Type: application/x-www-form-urlencoded; charset=utf-8
Date: Thu, 22 Feb 2018 07:46:12 GMT
x-acs-signature-nonce: 550e8400-e29b-41d4-a716-446655440000
x-acs-signature-method: HMAC-SHA1
x-acs-signature-version: 1.0
x-acs-version: 2016-01-02
```

#### 示例：规范请求

```
POST
application/json
ChDfdfwC+Tn874znq7Dw7Q==
application/x-www-form-urlencoded; charset=utf-8
Thu, 22 Feb 2018 07:46:12 GMT
x-acs-signature-nonce: 550e8400-e29b-41d4-a716-446655440000
x-acs-signature-method:HMAC-SHA1
x-acs-signature-version:1.0
x-acs-version:2016-01-02
/stacks?name=test_alert&status=COMPLETE
```

## 2. 添加签名。

将计算好的签名以如下格式添加到请求的Header中：

```
Authorization: acs AccessKeyId:Signature
```

### HTTP协议Header

计算签名必须包含一下参数，并按字典顺序排列；若值不存在则以“\n”补齐。

- Accept：客户端需要的返回值类型，取值：application/json | application/xml
- Content-MD5：HTTP协议消息体的128-bit MD5散列值转换成BASE64编码的结果。
- Content-Type：RFC 2616中定义的HTTP请求内容类型。
- Date：HTTP 1.1协议中规定的GMT时间，例如：Wed, 05 Sep. 2012 23:00:00 GMT。



说明：

不包含key。

#### 示例：原始header

```
Accept: application/json
```

```
Content-MD5: ChDfdfwC+Tn874znq7Dw7Q==
Content-Type: application/x-www-form-urlencoded;charset=utf-8
Date: Thu, 22 Feb 2018 07:46:12 GMT
```

#### 示例：规范header

```
application/json
ChDfdfwC+Tn874znq7Dw7Q==
application/x-www-form-urlencoded;charset=utf-8
Thu, 22 Feb 2018 07:46:12 GMT
```

### 阿里云协议Header (CanonicalizedHeaders)

阿里云规范头，非标准HTTP头部信息，是请求中出现的以`x-acs-`为前缀的参数。请求中必须包含以下参数：

- `x-acs-signature-nonce`：唯一随机数，用于防止网络重放攻击。在不同请求间要使用不同的随机数值。
- `x-acs-signature-version`：签名版本，取值：1.0
- `x-acs-version`：API版本号，请参照各产品的API文档。

完成以下操作，构造阿里云规范头：

1. 将所有以`x-acs-`为前缀的HTTP请求头的名字转换成小写字母。如将`X-acs-OSS-Meta-Name: TaoBao`转换成`x-acs-oss-meta-name: TaoBao`。
2. 将上一步得到的所有HTTP阿里云规范头按照字典序进行升序排列。
3. 删除请求头和内容之间分隔符两端出现的任何空格。如将`x-acs-oss-meta-name: TaoBao, Alipay`转换成`x-acs-oss-meta-name:TaoBao,Alipay`。
4. 将所有的头和内容用“\n”分隔符分隔拼成最后的CanonicalizedHeaders。

#### 示例：原始header

```
x-acs-signature-nonce: 550e8400-e29b-41d4-a716-446655440000
x-acs-signature-method: HMAC-SHA1
x-acs-signature-version: 1.0
x-acs-version: 2016-01-02GMT
```

#### 示例：规范header

```
x-acs-signature-nonce:550e8400-e29b-41d4-a716-446655440000
x-acs-signature-method:HMAC-SHA1
x-acs-signature-version:1.0
x-acs-version:2016-01-02
```

### 规范资源 (CanonicalizedResource)

CanonicalizedResource表示想要访问资源的规范描述，需要将子资源和query参数一同按照字典序，从小到大排列并以“&”为分隔符生成子资源字符串(?后的所有参数)。

**示例：原始请求**

```
/stacks?status=COMPLETE&name=test_alert
```

**示例：规范请求**

```
/stacks?name=test_alert&status=COMPLETE
```

**Body**

将请求的body用MD5算法加密，在进行base64编码，将结果添加到Content-MD5中。



## 5 公共请求和返回头

本文列举了轻量级分布式应用服务SAE的API的公共请求头和公共返回头信息。

### 公共请求头部

公共请求参数是指每个接口都需要使用到的请求参数。

参数名称	示例	是否必选	说明
Authorization	acs <yourAccessKeyId>:<yourSignature>	是	用于验证请求合法性的认证信息，采用 <code>AccessKeyId:Signature</code> 的形式。
Content-Length	0	是	RFC 2616 中定义的 HTTP 请求内容长度。
Content-Type	application/json	是	RFC 2616 中定义的 HTTP 请求内容类型。
Content-MD5	0e30656xxx xxxxxx0bc6 f70bbdfe	是	HTTP 协议消息体的 128-bit MD5 散列值转换成 BASE64 编码的结果。为了防止所有请求被篡改，建议所有请求都附加该信息。
Date	Mon, 26 Aug 2019 08:55:56 GMT	是	请求的构造时间，目前只支持 GMT 格式。如果与 MNS 的服务器时间前后差异超过 15 分钟将返回本次请求非法。
Host	sae.cn-beijing. aliyuncs.com	是	请求的服务地址，例如： <code>sae.cn-hangzhou.aliyuncs.com</code> 。
Accept	application/ json	是	客户端需要的返回值类型，支持 <code>application/json</code> 和 <code>application/xml</code> 。
x-acs-version	1.0	是	API 版本号。目前版本号为 2019-05-06。
x-acs-region-id	cn-beijing	是	地域ID。
x-acs-signature-nonce	f63659d4-10ac -483b-99da- ea8fde61eae3	是	唯一随机数，用于防止网络重放攻击。您在不同请求间要使用不同的随机数值。

参数名称	示例	是否必选	说明
x-acs-signature-method	HMAC-SHA1	是	用户签名方式，目前只支持 HMAC-SHA1。

### 示例

```
GET /pop/v1/paas/regionConfig HTTP/1.1
Host: sae.cn-beijing.aliyuncs.com
Accept: application/json
User-Agent: cs-sdk-python/0.0.1 (Darwin/15.2.0/x86_64;2.7.10)
x-acs-signature-nonce: f63659d4-10ac-483b-99da-ea8fde61eae3
Authorization: acs <yourAccessKeyId>:<yourSignature>
x-acs-signature-version: 1.0
Date: Mon, 26 Aug 2019 08:55:56 GMT
x-acs-signature-method: HMAC-SHA1
Content-Type: application/json;charset=utf-8
X-Acs-Region-Id: cn-beijing
Content-Length: 0
```

### 公共返回头部

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。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；返回的HTTP状态码为 4xx 或 5xx，代表调用失败。

### XML 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 示例:

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

## 6 命名空间和VPC

### 6.1 CreateNamespace

创建命名空间

调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

请求头

该接口使用公共请求头，无特殊请求头。请参见公共请求参数文档。

请求语法

```
POST /pop/v1/paas/namespace HTTP/1.1
```

请求参数

名称	类型	是否必选	示例值	描述
NamespaceId	String	是	cn-beijing:test	命名空间ID
NamespaceName	String	是	name	命名空间名称
NamespaceDescription	String	否	description	描述

返回数据

名称	类型	示例值	描述
Code	String	200	接口状态或 POP 错误码
Data			命名空间信息
NamespaceDescription	String	desc	描述
NamespaceId	String	cn-beijing:test	命名空间ID

名称	类型	示例值	描述
NamespaceName	String	name	命名空间名称
RegionId	String	cn-beijing	地域
ErrorCode	String	success	错误码
Message	String	success	附加信息
RequestId	String	00000000-0000-0000-0000-000000000000	请求ID
Success	Boolean	true	是否成功
TraceId	String	0000000000 0000000000 0000000000	调用链ID

## 示例

### 请求示例

```
POST /pop/v1/paas/namespace HTTP/1.1
公共请求头
{
  "NamespaceId": "cn-beijing:test"
  "NamespaceName": "name"
}
```

### 正常返回示例

#### XML 格式

```
<CreateNamespaceResponse>
  <Data>
    <RegionId>cn-beijing</RegionId>
    <NamespaceName>name</NamespaceName>
    <NamespaceDescription>desc</NamespaceDescription>
    <NamespaceId>cn-beijing:test</NamespaceId>
  </Data>
  <Message>success</Message>
  <RequestId>00000000-0000-0000-0000-000000000000</RequestId>
  <TraceId>00000000000000000000000000000000</TraceId>
  <Success>>true</Success>
  <ErrorCode>success</ErrorCode>
  <Code>200</Code>
```

```
</CreateNamespaceResponse>
```

## JSON 格式

```
{
  "Data":{
    "NamespaceName":"name",
    "RegionId":"cn-beijing",
    "NamespaceDescription":"desc",
    "NamespaceId":"cn-beijing:test"
  },
  "Message":"success",
  "RequestId":"000000000-00000-00000-00000-00000000000000",
  "TraceId":"0000000000000000000000000000000000",
  "ErrorCode":"success",
  "Success":true,
  "Code":200
}
```

## 错误码

HttpCode	错误码	错误信息	描述
400	InstanceExist.NamespaceId	The specified namespace ID already exists.	指定的命名空间ID已存在
400	InvalidNamespace.WithUppercase	This namespace does not support creating SAE apps because it contains uppercase letters.	命名空间不支持创建SAE应用，因为它带有大写字母。
400	InvalidNamespaceId.Format	The specified NamespaceId is invalid.	指定的NamespaceId不合法。正确格式为 [所属regionId]:[命名空间标识]，例如 cn-beijing:test，并且长度不能超过32个字符。
400	InvalidNamespaceIdSuffix.Format	The specified NamespaceId is invalid. NamespaceId can only contain alphabetical letters or numbers.	指定的NamespaceId不合法。命名空间ID只能由字母或数字组成。
400	InvalidNamespaceName.Format	The specified NamespaceName is invalid. The name of the namespace cannot exceed 63 characters in length.	指定的NamespaceName不合法。命名空间名称的长度不能超过63个字符。
400	InvalidOperation.NamespaceClusterNotDeleted	The specified NamespaceId contains clusters.	指定的NamespaceId下还有集群。

HttpCode	错误码	错误信息	描述
400	Namespace.AppExists	Please delete the application first.	请先删除应用。

访问[错误中心](#)查看更多错误码。

## 6.2 DeleteNamespace

删除命名空间

调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

请求头

该接口使用公共请求头，无特殊请求头。请参见公共请求参数文档。

请求语法

```
DELETE /pop/v1/paas/namespace HTTP/1.1
```

请求参数

名称	类型	是否必选	示例值	描述
NamespaceId	String	是	cn-beijing:test	命名空间ID

返回数据

名称	类型	示例值	描述
Code	String	200	接口状态或 POP 错误码
ErrorCode	String	success	错误码
Message	String	success	附加信息
RequestId	String	00000000-0000-0000-0000-000000000000	请求ID
Success	Boolean	true	是否成功

名称	类型	示例值	描述
TraceId	String	0000000000 0000000000 0000000000	调用链ID

## 示例

### 请求示例

```
DELETE /pop/v1/paas/namespace HTTP/1.1
公共请求头
{
  "NamespaceId": "cn-beijing:test"
}
```

### 正常返回示例

#### XML 格式

```
<DeleteNamespaceResponse>
  <Message>success</Message>
  <RequestId>000000000-0000-0000-0000-000000000000</RequestId>
  <TraceId>00000000000000000000000000000000</TraceId>
  <ErrorCode>success</ErrorCode>
  <Success>>true</Success>
  <Code>200</Code>
</DeleteNamespaceResponse>
```

#### JSON 格式

```
{
  "Message": "success",
  "RequestId": "000000000-0000-0000-0000-000000000000",
  "TraceId": "00000000000000000000000000000000",
  "Success": true,
  "ErrorCode": "success",
  "Code": 200
}
```

## 错误码

HttpCode	错误码	错误信息	描述
404	InvalidNamespaceId.NotFound	The specified NamespaceId does not exist.	指定的NamespaceId不存在。
400	InvalidOperation.NamespaceClusterNotDeleted	The specified NamespaceId contains clusters.	指定的NamespaceId下还有集群。
400	Namespace.AppExists	Please delete the application first.	请先删除应用。

访问[错误中心](#)查看更多错误码。

## 6.3 DescribeNamespace

查询命名空间详细信息

调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

请求头

该接口使用公共请求头，无特殊请求头。请参见公共请求参数文档。

请求语法

```
GET /pop/v1/paas/namespace HTTP/1.1
```

请求参数

名称	类型	是否必选	示例值	描述
NamespaceId	String	是	cn-beijing:test	命名空间ID

返回数据

名称	类型	示例值	描述
Code	String	200	接口状态或 POP 错误码
Data			命名空间信息
NamespaceDescription	String	desc	命名空间描述
NamespaceId	String	cn-beijing:test	命名空间ID
NamespaceName	String	name	命名空间名称
RegionId	String	cn-beijing	地域
ErrorCode	String	success	错误码
Message	String	success	附加信息



名称	类型	示例值	描述
RequestId	String	00000000-0000-0000-0000-000000000000	请求ID
Success	Boolean	true	是否成功
TraceId	String	0000000000 0000000000 0000000000	调用链ID

## 示例

### 请求示例

```
GET /pop/v1/paas/namespace HTTP/1.1
公共请求头
{
  "NamespaceId": "cn-beijing:test"
}
```

### 正常返回示例

#### XML 格式

```
<DescribeNamespaceResponse>
  <Data>
    <NamespaceName>test</NamespaceName>
    <RegionId>cn-beijing</RegionId>
    <NamespaceDescription>test</NamespaceDescription>
    <NamespaceId>cn-beijing:test</NamespaceId>
  </Data>
  <Message>success</Message>
  <RequestId>00000000-0000-0000-0000-000000000000</RequestId>
  <TraceId>00000000000000000000000000000000</TraceId>
  <ErrorCode>success</ErrorCode>
  <Success>>true</Success>
  <Code>200</Code>
</DescribeNamespaceResponse>
```

#### JSON 格式

```
{
  "Data":{
    "NamespaceName":"test",
    "RegionId":"cn-beijing",
    "NamespaceDescription":"test",
    "NamespaceId":"cn-beijing:test"
  },
  "Message":"success",
  "RequestId":"00000000-0000-0000-0000-000000000000",
  "TraceId":"00000000000000000000000000000000",
  "Success":true,
  "ErrorCode":"success",
  "Code":200
}
```

```
}

```

### 错误码

访问[错误中心](#)查看更多错误码。

## 6.4 DescribeNamespaces

查询命名空间列表

### 调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

### 请求头

该接口使用公共请求头，无特殊请求头。请参见公共请求参数文档。

### 请求语法

```
GET /pop/v1/paas/namespaces HTTP/1.1

```

### 请求参数

名称	类型	是否必选	示例值	描述
CurrentPage	Integer	是	1	页码
PageSize	Integer	是	10	翻页大小

### 返回数据

名称	类型	示例值	描述
Code	String	200	接口状态或 POP 错误码
Data			命名空间信息列表
CurrentPage	Integer	1	页码
Namespaces			命名空间列表
AccessKey	String	xxx	AK
NamespaceDescription	String	desc	命名空间描述

名称	类型	示例值	描述
NamespaceId	String	cn-beijing:test	命名空间ID
NamespaceName	String	name	命名空间名称
RegionId	String	cn-beijing	地域
SecretKey	String	xxx	SK
TenantId	String	00000000-0000-0000-0000-000000000000	租户ID
PageSize	Integer	10	翻页大小
TotalSize	Integer	100	总数
ErrorCode	String	success	错误码
Message	String	success	附加信息
RequestId	String	00000000-0000-0000-0000-000000000000	请求ID
Success	Boolean	true	是否成功
TraceId	String	0000000000 0000000000 0000000000	调用链ID

## 示例

### 请求示例

```
GET /pop/v1/paas/namespaces HTTP/1.1
公共请求头
{
  "CurrentPage": "1"
  "PageSize": "10"
}
```

### 正常返回示例

## XML 格式

```
<DescribeNamespacesResponse>
  <Data>
    <Namespaces>
      <TenantId>00000000-0000-0000-0000-000000000000</TenantId>
      <NamespaceName>Default</NamespaceName>
      <AccessKey>xxx</AccessKey>
      <RegionId>cn-beijing</RegionId>
      <SecretKey>xxx</SecretKey>
      <NamespaceDescription>Default Namespace</NamespaceDescription>
    >
  </Namespaces>
  <PageSize>1</PageSize>
  <CurrentPage>1</CurrentPage>
  <TotalSize>1</TotalSize>
</Data>
<Message>success</Message>
<RequestId>00000000-0000-0000-0000-000000000000</RequestId>
<TraceId>00000000000000000000000000000000</TraceId>
<ErrorCode>success</ErrorCode>
<Success>>true</Success>
<Code>200</Code>
</DescribeNamespacesResponse>
```

## JSON 格式

```
{
  "Data": {
    "Namespaces": [
      {
        "TenantId": "00000000-0000-0000-0000-000000000000",
        "NamespaceName": "Default",
        "AccessKey": "xxx",
        "RegionId": "cn-beijing",
        "SecretKey": "xxx",
        "NamespaceDescription": "Default Namespace"
      }
    ],
    "PageSize": 1,
    "TotalSize": 1,
    "CurrentPage": 1
  },
  "Message": "success",
  "RequestId": "00000000-0000-0000-0000-000000000000",
  "TraceId": "00000000000000000000000000000000",
  "Success": true,
  "ErrorCode": "success",
  "Code": 200
}
```

## 错误码

访问[错误中心](#)查看更多错误码。

## 6.5 UpdateNamespace

更新命名空间信息

调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

请求头

该接口使用公共请求头，无特殊请求头。请参见公共请求参数文档。

请求语法

```
PUT /pop/v1/paas/namespace HTTP/1.1
```

请求参数

名称	类型	是否必选	示例值	描述
NamespaceId	String	是	cn-beijing:test	命名空间ID
NamespaceName	String	是	name	命名空间名称
NamespaceDescription	String	否	desc	命名空间描述

返回数据

名称	类型	示例值	描述
Code	String	200	接口状态或 POP 错误码
Data			命名空间信息
NamespaceDescription	String	desc	命名空间描述
NamespaceId	String	cn-beijing:test	命名空间ID
NamespaceName	String	name	命名空间名称
RegionId	String	cn-beijing	地域

名称	类型	示例值	描述
ErrorCode	String	success	错误码
Message	String	success	附加信息
RequestId	String	00000000-0000-0000-0000-000000000000	请求ID
Success	Boolean	true	是否成功
TraceId	String	0000000000 0000000000 0000000000	调用链ID

## 示例

### 请求示例

```
PUT /pop/v1/paas/namespace HTTP/1.1
公共请求头
{
  "NamespaceId": "cn-beijing:test"
  "NamespaceName": "name"
}
```

### 正常返回示例

#### XML 格式

```
<UpdateNamespaceResponse>
  <Data>
    <NamespaceName>name</NamespaceName>
    <RegionId>cn-beijing</RegionId>
    <NamespaceDescription>desc</NamespaceDescription>
    <NamespaceId>cn-beijing:test</NamespaceId>
  </Data>
  <Message>success</Message>
  <RequestId>00000000-0000-0000-0000-000000000000</RequestId>
  <TraceId>00000000000000000000000000000000</TraceId>
  <Success>true</Success>
  <ErrorCode>success</ErrorCode>
  <Code>200</Code>
</UpdateNamespaceResponse>
```

#### JSON 格式

```
{
  "Data":{
    "NamespaceName":"name",
    "RegionId":"cn-beijing",
    "NamespaceDescription":"desc",
    "NamespaceId":"cn-beijing:test"
```

```

    },
    "Message": "success",
    "RequestId": "000000000-0000-0000-0000-0000000000000",
    "TraceId": "00000000000000000000000000000000",
    "ErrorCode": "success",
    "Success": true,
    "Code": 200
  }
}

```

### 错误码

HttpCode	错误码	错误信息	描述
404	InvalidNamespaceId.NotFound	The specified NamespaceId does not exist.	指定的NamespaceId不存在。
400	InvalidNamespaceName.Format	The specified NamespaceName is invalid. The name of the namespace cannot exceed 63 characters in length.	指定的NamespaceName不合法。命名空间名称的长度不能超过63个字符。

访问[错误中心](#)查看更多错误码。

## 6.6 DescribeNamespaceList

获取命名空间列表

### 调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

### 请求头

该接口使用公共请求头，无特殊请求头。请参见公共请求参数文档。

### 请求语法

```
GET /pop/v1/sam/namespace/describeNamespaceList HTTP/1.1
```

### 请求参数

名称	类型	是否必选	示例值	描述
ContainCustom	Boolean	是	true	是否返回自定义命名空间
HybridCloudExclude	Boolean	否	true	是否排除混合云命名空间

## 返回数据

名称	类型	示例值	描述
Code	String	200	接口状态
Data			命名空间列表
AgentInstall	String	http://edas-bj.oss-cn-beijing-internal.aliyuncs.com/test/install.sh	EDAS Agent安装命令
Current	Boolean	false	已废弃
Custom	Boolean	true	自定义命名空间
HybridCloudEnable	Boolean	false	混合云命名空间
NamespaceId	String	cn-beijing:test	命名空间ID
NamespaceName	String	test	命名空间名称
RegionId	String	cn-beijing	命名空间归属的Region
VSwitchId	String	vsw-2ze559r1z1bpwqxwpa8o9	VSwitch ID
VpcId	String	vpc-2ze0i263cn311nvji7jj	VPC ID
ErrorCode	String	success	错误码
Message	String	success	附加信息
RequestId	String	30375C38-F4ED-4135-A0AE-5C75DC7F7B3F	请求ID
Success	Boolean	true	是否成功标识
TraceId	String	ac1a0b2215622920113732501e6d56	调用ID



## 示例

### 请求示例

```
http(s)://[Endpoint]/?<公共请求参数>
```

### 正常返回示例

#### XML 格式

```
<DescribeNamespaceListResponse>
  <Message>success</Message>
  <RequestId>30375C38-F4ED-4135-A0AE-5C75DC7F7B3F</RequestId>
  <TraceId>ac1a0b2215622920113732501e6d56</TraceId>
  <Data>
    <NamespaceName>华北2</NamespaceName>
    <VpcId>vpc-2ze0i263cnn311nvji7jj</VpcId>
    <VSwitchId>vsw-2ze559r1z1bpwqxwpa8o9</VSwitchId>
    <AgentInstall>http://edas-bj.oss-cn-beijing-internal.aliyuncs.com/test/install.sh</AgentInstall>
    <Custom>>false</Custom>
    <RegionId>cn-beijing</RegionId>
    <NamespaceId>cn-beijing</NamespaceId>
    <Current>>false</Current>
    <HybridCloudEnable>>false</HybridCloudEnable>
  </Data>
  <Data>
    <NamespaceName>xxx</NamespaceName>
    <AgentInstall>http://edas-qd.oss-cn-qingdao-internal.aliyuncs.com/test/install.sh</AgentInstall>
    <Custom>>true</Custom>
    <RegionId>cn-beijing</RegionId>
    <NamespaceId>cn-beijing:xxx</NamespaceId>
    <Current>>false</Current>
    <HybridCloudEnable>>false</HybridCloudEnable>
  </Data>
  <ErrorCode>success</ErrorCode>
  <Code>200</Code>
  <Success>>true</Success>
</DescribeNamespaceListResponse>
```

#### JSON 格式

```
{
  "Data": [
    {
      "AgentInstall": "http://edas-bj.oss-cn-beijing-internal.aliyuncs.com/test/install.sh",
      "NamespaceName": "华北2",
      "RegionId": "cn-beijing",
      "HybridCloudEnable": false,
      "Custom": false,
      "VSwitchId": "vsw-2ze559r1z1bpwqxwpa8o9",
      "VpcId": "vpc-2ze0i263cnn311nvji7jj",
      "Current": false,
      "NamespaceId": "cn-beijing"
    },
    {

```

```
    "AgentInstall": "http://edas-qd.oss-cn-qingdao-internal.aliyuncs.com/test/install.sh",
    "NamespaceName": "xxx",
    "RegionId": "cn-beijing",
    "HybridCloudEnable": false,
    "Custom": true,
    "Current": false,
    "NamespaceId": "cn-beijing:xxx"
  }
],
"Message": "success",
"RequestId": "30375C38-F4ED-4135-A0AE-5C75DC7F7B3F",
"TraceId": "ac1a0b2215622920113732501e6d56",
"Success": true,
"ErrorCode": "success",
"Code": 200
}
```

## 错误码

访问[错误中心](#)查看更多错误码。

## 7 应用信息

### 7.1 DescribeApplicationConfig

获取应用配置信息。

调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

请求头

该接口使用公共请求头，无特殊请求头。请参见公共请求参数文档。

请求语法

```
GET /pop/v1/sam/app/describeApplicationConfig HTTP/1.1
```

请求参数

名称	类型	是否必选	示例值	描述
AppId	String	否	7171a6ca-d1cd-4928-8642-7d5cfe69a26c	应用ID。

返回数据

名称	类型	示例值	描述
Code	String	200	接口状态
Data			应用信息
AppDescription	String	示例应用	应用描述信息
AppId	String	7171a6ca-d1cd-4928-8642-7d5cfe69a26c	应用ID
AppName	String	demo-app	应用名称
BatchWaitTime	Integer	10	分批发布时批次间的等待时间（单位秒）

名称	类型	示例值	描述
Command	String	/bin/bash	应用容器启动命令
CommandArgs	String	["-c","echo 1234"]	应用容器启动命令参数
Cpu	Integer	1000	应用CPU规格（单位豪核，1核=1000豪核）
CustomHostAlias	String	{ "hostName": "test.host.name", "ip": "0.0.0.0" }	Hosts绑定设置
EdasContainerVersion	String	3.5.3	EDAS应用环境版本
Envs	String	{ "name": "TEST_ENV_KEY", "value": "TEST_ENV_VAR" }	环境变量
ImageUrl	String	docker.io/library/nginx:1.14.2	镜像地址
JarStartArgs	String	start	Jar包部署应用的args设置
JarStartOptions	String	-Dtest=true	Jar包部署应用的options设置
Jdk	String	Open JDK 8	应用使用的JDK版本信息
Liveness	String	{ "exec": { "command": ["curl http://localhost:8080"], "initialDelaySeconds": 20, "timeoutSeconds": 3 } }	应用实例存活检查(Liveness配置)
Memory	Integer	2048	应用内存规格（单位MB）
MinReadyInstances	Integer	1	最小存活实例数
NamespaceId	String	cn-beijing:test	EDAS命名空间ID

名称	类型	示例值	描述
PackageType	String	War	应用部署类型 (Image, War, Jar)
PackageUrl	String	https://edas-bj.oss-cn-beijing.aliyuncs.com/apps/K8S_APP_ID/d4c97c37-aba3-403e-ae1e-6f7d87423016/hello-edas.war	应用部署包地址
PackageVersion	String	1.0	应用部署包版本
Readiness	String	{"exec":{"command":["curl http://localhost:8080"],"initialDelaySeconds":20,"timeoutSeconds":5}}	应用业务就绪检查(Readiness配置)
Replicas	Integer	2	应用实例个数
ErrorCode	String	success	错误码
Message	String	success	附加信息
RequestId	String	01CF26C7-00A3-4AA6-BA76-7E95F2A3B6FF	请求ID
Success	Boolean	true	是否成功标识
TraceId	String	ac1a0b2215622246421415014e6d54	调用链ID

## 示例

### 请求示例

```
GET /pop/v1/sam/app/describeApplicationConfig HTTP/1.1
```

## 公共请求头

### 正常返回示例

#### XML 格式

```
<DescribeApplicationConfigResponse>
  <Message>success</Message>
  <RequestId>01CF26C7-00A3-4AA6-BA76-7E95F2A3B6FF</RequestId>
  <TraceId>ac1a0b2215622246421415014e6d54</TraceId>
  <Data>
    <AppDescription></AppDescription>
    <Liveness></Liveness>
    <Memory>2048</Memory>
    <WebContainer>apache-tomcat-7.0.91</WebContainer>
    <Cpu>1000</Cpu>
    <PackageVersion>222</PackageVersion>
    <AppName>demo-app</AppName>
    <Jdk>Open JDK 8</Jdk>
    <JarStartArgs></JarStartArgs>
    <MinReadyInstances>0</MinReadyInstances>
    <Readiness></Readiness>
    <PackageType>War</PackageType>
    <CommandArgs></CommandArgs>
    <VSwitchId>vsw-xxxxx</VSwitchId>
    <Envs>[{"name":"TEST_ENV_KEY","value":"TEST_ENV_VAR"}]</Envs>
    <JarStartOptions></JarStartOptions>
    <Replicas>1</Replicas>
    <CustomHostAlias>[{"hostName":"test.host.name","ip":"0.0.0.0"}]</CustomHostAlias>
    <AppId>7171a6ca-d1cd-4928-8642-7d5cfe69a26c</AppId>
    <VpcId>vpc-xxxx</VpcId>
    <Command></Command>
    <PackageUrl>https://edas-bj.oss-cn-beijing.aliyuncs.com/hello-edas.war</PackageUrl>
    <BatchWaitTime>10</BatchWaitTime>
    <NamespaceId>cn-beijing</NamespaceId>
    <RegionId>cn-beijing</RegionId>
  </Data>
  <ErrorCode>success</ErrorCode>
  <Code>200</Code>
  <Success>>true</Success>
</DescribeApplicationConfigResponse>
```

#### JSON 格式

```
{
  "Data": {
    "Command": "",
    "MinReadyInstances": 0,
    "Readiness": "",
    "AppId": "7171a6ca-d1cd-4928-8642-7d5cfe69a26c",
    "VSwitchId": "vsw-xxxxx",
    "VpcId": "vpc-xxxx",
    "Cpu": 1000,
    "Memory": 2048,
    "PackageType": "War",
    "CommandArgs": "",
    "AppDescription": "",
    "AppName": "demo-app",
    "NamespaceId": "cn-beijing",
```

```

    "Jdk":"Open JDK 8",
    "JarStartOptions":"","
    "PackageVersion":"222",
    "CustomHostAlias":["{"hostname":"test.host.name","ip":"0.0.0.
0"}"],
    "JarStartArgs":"","
    "Envs":["{"name":"TEST_ENV_KEY","value":"TEST_ENV_VAR"}"],
    "RegionId":"cn-beijing",
    "PackageUrl":"https://edas-bj.oss-cn-beijing.aliyuncs.com/hello-edas
.war",
    "WebContainer":"apache-tomcat-7.0.91",
    "Liveness":"","
    "Replicas":1,
    "BatchWaitTime":10
  },
  "Message":"success",
  "RequestId":"01CF26C7-00A3-4AA6-BA76-7E95F2A3B6FF",
  "TraceId":"ac1a0b2215622246421415014e6d54",
  "Success":true,
  "ErrorCode":"success",
  "Code":200
}

```

### 错误码

访问[错误中心](#)查看更多错误码。

## 7.2 DescribeRegions

使用 DescribeRegions 查询可用地域。

### 调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

### 请求头

该接口使用公共请求头，无特殊请求头。请参见公共请求参数文档。

### 请求语法

```
GET /pop/v1/paas/regionConfig HTTP/1.1
```

### 请求参数

### 返回数据

名称	类型	示例值	描述
Code	Integer	200	接口状态
Message	String	success	附加信息

名称	类型	示例值	描述
Regions			地域列表
RequestId	String	DDE85827-B0B3-4E56-86E8-17C420090D5C	请求 ID

## 示例

### 请求示例

```
GET /pop/v1/paas/regionConfig HTTP/1.1  
公共请求头
```

### 正常返回示例

#### XML 格式

```
<DescribeRegionsResponse>  
  <Message>success</Message>  
  <RequestId>DDE85827-B0B3-4E56-86E8-17C420090D5C</RequestId>  
  <Regions>  
    <Region>  
      <RegionId>cn-shanghai</RegionId>  
      <RegionEndpoint>sae.cn-shanghai.aliyuncs.com</RegionEndpoint>  
      <LocalName>华东2（上海）</LocalName>  
    </Region>  
    <Region>  
      <RegionId>cn-hangzhou</RegionId>  
      <RegionEndpoint>sae.cn-hangzhou.aliyuncs.com</RegionEndpoint>  
      <LocalName>华东1（杭州）</LocalName>  
    </Region>  
  </Regions>  
  <Code>200</Code>  
</DescribeRegionsResponse>
```

#### JSON 格式

```
{  
  "Message": "success",  
  "RequestId": "DDE85827-B0B3-4E56-86E8-17C420090D5C",  
  "Regions": {  
    "Region": [  
      {  
        "RegionId": "cn-shanghai",  
        "RegionEndpoint": "sae.cn-shanghai.aliyuncs.com",  
        "LocalName": "华东2（上海）"  
      },  
      {  
        "RegionId": "cn-hangzhou",  
        "RegionEndpoint": "sae.cn-hangzhou.aliyuncs.com",  
        "LocalName": "华东1（杭州）"  
      }  
    ]  
  },  
  "Code": 200  
}
```



```
}

```

## 错误码

访问[错误中心](#)查看更多错误码。

## 7.3 DescribeInstanceLog

获取实例日志

### 调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

### 请求头

该接口使用公共请求头，无特殊请求头。请参见公共请求参数文档。

### 请求语法

```
GET /pop/v1/sam/instance/describeInstanceLog HTTP/1.1

```

### 请求参数

名称	类型	是否必选	示例值	描述
InstanceId	String	是	xxx	实例ID

### 返回数据

名称	类型	示例值	描述
Code	String	200	接口状态或 POP 错误码
Data	String	hello\nedas\n	实例日志信息
ErrorCode	String	success	错误码
Message	String	success	附加信息
RequestId	String	00000000-0000-0000-0000-000000000000	请求ID
Success	Boolean	true	是否成功

名称	类型	示例值	描述
TraceId	String	0000000000 0000000000 0000000000	调用链ID

## 示例

### 请求示例

```
GET /pop/v1/sam/instance/describeInstanceLog HTTP/1.1
公共请求头
{
  "InstanceId": "xxx"
}
```

### 正常返回示例

#### XML 格式

```
<DescribeInstanceLogResponse>
  <Data>hello\nsae\n</Data>
  <Message>success</Message>
  <RequestId>00000000-0000-0000-0000-000000000000</RequestId>
  <TraceId>00000000000000000000000000000000</TraceId>
  <ErrorCode>success</ErrorCode>
  <Success>true</Success>
  <Code>200</Code>
</DescribeInstanceLogResponse>
```

#### JSON 格式

```
{
  "Data": "hello\nsae\n",
  "Message": "success",
  "RequestId": "00000000-0000-0000-0000-000000000000",
  "TraceId": "00000000000000000000000000000000",
  "Success": true,
  "ErrorCode": "success",
  "Code": 200
}
```

## 错误码

访问[错误中心](#)查看更多错误码。

## 7.4 DescribeComponents

获取应用创建部署时所需的组件版本

调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

请求头

该接口使用公共请求头，无特殊请求头。请参见公共请求参数文档。

请求语法

```
GET /pop/v1/sam/resource/components HTTP/1.1
```

请求参数

名称	类型	是否必选	示例值	描述
AppId	String	是	00000000-0000-0000-0000-000000000000	应用ID
Type	String	是	TOMCAT	支持的组件类型。 · Tomcat · JDK

返回数据

名称	类型	示例值	描述
Code	String	200	接口状态或 POP 错误码
Data			支持的应用组件信息
ComponentDescription	String	Open JDK 8	组件描述
ComponentKey	String	Open JDK 8	组件ID
Expired	Boolean	false	是否过期
Type	String	JDK	组件类型

名称	类型	示例值	描述
ErrorCode	String	success	错误码
Message	String	success	附加信息
RequestId	String	00000000-0000-0000-0000-000000000000	请求ID
Success	Boolean	true	是否成功
TraceId	String	0000000000 0000000000 0000000000	调用链ID

示例

请求示例

```
GET /pop/v1/sam/resource/components HTTP/1.1
公共请求头
{
  "AppId": "00000000-0000-0000-0000-000000000000"
  "Type": "TOMCAT"
}
```

正常返回示例

XML 格式

```
<DescribeComponentsResponse>
  <Data>
    <ComponentKey>Open JDK 8</ComponentKey>
    <Type>JDK</Type>
    <Expired>>false</Expired>
    <ComponentDescription>Open JDK 8</ComponentDescription>
  </Data>
  <Data>
    <ComponentKey>Open JDK 7</ComponentKey>
    <Type>JDK</Type>
    <Expired>>false</Expired>
    <ComponentDescription>Open JDK 7</ComponentDescription>
  </Data>
  <Message>success</Message>
  <RequestId>00000000-0000-0000-0000-000000000000</RequestId>
  <TraceId>00000000000000000000000000000000</TraceId>
  <Success>>true</Success>
  <ErrorCode>success</ErrorCode>
  <Code>200</Code>
```

```
</DescribeComponentsResponse>
```

### JSON 格式

```
{
  "Data": [
    {
      "ComponentKey": "Open JDK 8",
      "Type": "JDK",
      "ComponentDescription": "Open JDK 8",
      "Expired": false
    },
    {
      "ComponentKey": "Open JDK 7",
      "Type": "JDK",
      "ComponentDescription": "Open JDK 7",
      "Expired": false
    }
  ],
  "Message": "success",
  "RequestId": "000000000-0000-0000-0000-000000000000",
  "TraceId": "00000000000000000000000000000000",
  "ErrorCode": "success",
  "Success": "true",
  "Code": 200
}
```

### 错误码

访问[错误中心](#)查看更多错误码。

## 7.5 DescribeEdasContainers

获取应用微服务容器组件列表

### 调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

### 请求头

该接口使用公共请求头，无特殊请求头。请参见公共请求参数文档。

## 请求语法

```
GET /pop/v1/sam/resource/edasContainers HTTP/1.1
```

## 请求参数

## 返回数据

名称	类型	示例值	描述
Code	String	200	接口状态或 POP 错误码
Data			组件列表
Disabled	Boolean	false	是否禁用
EdasContainerVersion	String	3.5.3	容器版本
ErrorCode	String	success	错误码
Message	String	success	错误码
RequestId	String	00000000-0000-0000-0000-000000000000	请求ID
Success	Boolean	true	是否成功
TraceId	String	0000000000 0000000000 0000000000	调用链ID

## 示例

## 请求示例

```
GET /pop/v1/sam/resource/edasContainers HTTP/1.1  
公共请求头
```

## 正常返回示例

## XML 格式

```
<DescribeEdasContainersResponse>  
  <Data>  
    <EdasContainerVersion>3.5.3</EdasContainerVersion>  
    <Disabled>>false</Disabled>  
  </Data>  
  <Message>success</Message>
```

```
<RequestId>00000000-0000-0000-0000-000000000000</RequestId>
<TraceId>00000000000000000000000000000000</TraceId>
<Success>>true</Success>
<ErrorCode>success</ErrorCode>
<Code>200</Code>
</DescribeEdasContainersResponse>
```

### JSON 格式

```
{
  "Data": [
    {
      "Disabled": false,
      "EdasContainerVersion": "3.5.3"
    }
  ],
  "Message": "success",
  "RequestId": "00000000-0000-0000-0000-000000000000",
  "TraceId": "00000000000000000000000000000000",
  "ErrorCode": "success",
  "Success": true,
  "Code": 200
}
```

### 错误码

访问[错误中心](#)查看更多错误码。

## 7.6 DescribeApplicationImage

描述应用镜像信息

### 调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

### 请求头

该接口使用公共请求头，无特殊请求头。请参见公共请求参数文档。

### 请求语法

```
GET /pop/v1/sam/container/describeApplicationImage HTTP/1.1
```

### 请求参数

名称	类型	是否必选	示例值	描述
AppId	String	是	00000000-0000-0000-0000-000000000000	应用ID

名称	类型	是否必选	示例值	描述
ImageUrl	String	是	registry-vpc.cn-hangzhou.aliyuncs.com/demo/demo:latest	镜像Url

## 返回数据

名称	类型	示例值	描述
Code	String	200	接口状态或 POP 错误码
Data			应用镜像信息
CrUrl	String	保留字段	保留字段
Logo	String	保留字段	保留字段
RegionId	String	cn-beijing	地域
RepoName	String	demo	镜像仓库名称
RepoNamespace	String	demo	镜像命名空间
RepoOriginType	String	ALI_HUB	镜像仓库类型。当前仅支持容器镜像服务
RepoTag	String	latest	镜像TAG
RepoType	String	保留字段	保留字段
ErrorCode	String	success	错误码
Message	String	success	附加信息
RequestId	String	00000000-0000-0000-000000000000	请求ID
Success	Boolean	true	是否成功



名称	类型	示例值	描述
TraceId	String	0000000000 0000000000 0000000000	调用链ID

## 示例

### 请求示例

```
GET /pop/v1/sam/container/describeApplicationImage HTTP/1.1
公共请求头
{
  "AppId": "000000000-0000-0000-0000-000000000000"
  "ImageUrl": "registry-vpc.cn-hangzhou.aliyuncs.com/demo/demo:latest"
}
```

### 正常返回示例

#### XML 格式

```
<DescribeApplicationImageResponse>
  <Data>
    <RepoName>nginx</RepoName>
    <RepoOriginType>ALI_HUB</RepoOriginType>
    <RepoNamespace>demo</RepoNamespace>
    <RepoTag>latest</RepoTag>
    <RegionId>cn-hangzhou</RegionId>
  </Data>
  <Message>success</Message>
  <RequestId>00000000-0000-0000-0000-000000000000</RequestId>
  <TraceId>00000000000000000000000000000000</TraceId>
  <ErrorCode>success</ErrorCode>
  <Success>>true</Success>
  <Code>200</Code>
</DescribeApplicationImageResponse>
```

#### JSON 格式

```
{
  "Data":{
    "RepoName":"nginx",
    "RepoOriginType":"ALI_HUB",
    "RepoNamespace":"demo",
    "RegionId":"cn-hangzhou",
    "RepoTag":"latest"
  },
  "Message":"success",
  "RequestId":"00000000-0000-0000-0000-000000000000",
  "TraceId":"00000000000000000000000000000000",
  "Success":true,
  "ErrorCode":"success",
  "Code":200
}
```

```
}

```

### 错误码

访问[错误中心](#)查看更多错误码。

## 7.7 DescribeApplicationInstances

获取应用实例列表

### 调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

### 请求头

该接口使用公共请求头，无特殊请求头。请参见公共请求参数文档。

### 请求语法

```
GET /pop/v1/sam/app/describeApplicationInstances HTTP/1.1
```

### 请求参数

名称	类型	是否必选	示例值	描述
AppId	String	是	00000000-0000-0000-0000-000000000000	应用ID
GroupId	String	是	00000000-0000-0000-0000-000000000000	应用分组ID。请通过 DescribeApplicationGroups 接口获得
CurrentPage	Integer	否	1	页码
PageSize	Integer	否	10	页面大小

### 返回数据

名称	类型	示例值	描述
Code	String	200	接口状态或 POP 错误码
Data			应用实例信息
CurrentPage	Integer	1	页码

名称	类型	示例值	描述
Instances			应用实例列表
CreateTime Stamp	Long	1558442609000	实例创建时间戳，单位毫秒
GroupId	String	00000000-0000-0000-0000-000000000000	实例所属分组ID
GroupId	String	00000000-0000-0000-0000-000000000000	实例所属分组ID
InstanceContainerIp	String	192.168.1.1	实例内网IP
InstanceContainerStatus	String	Running	实例运行状态。如 Running 为已启动
InstanceId	String	00000000-0000-0000-0000-000000000000	实例ID
PageSize	Integer	10	页码大小
TotalSize	Integer	30	实例总数
ErrorCode	String	success	错误码
Message	String	success	附加信息
RequestId	String	00000000-0000-0000-0000-000000000000	请求ID
Success	Boolean	true	是否成功
TraceId	String	0000000000 0000000000 0000000000	调用链ID

## 示例

### 请求示例

```
GET /pop/v1/sam/app/describeApplicationInstances HTTP/1.1
公共请求头
```

```
{
  "AppId": "00000000-0000-0000-0000-000000000000"
  "GroupId": "00000000-0000-0000-0000-000000000000"
}
```

## 正常返回示例

### XML 格式

```
<DescribeApplicationInstancesResponse>
  <Data>
    <PageSize>1</PageSize>
    <Instances>
      <CreateTimeStamp>1558442609000</CreateTimeStamp>
      <InstanceContainerIp>192.168.1.1</InstanceContainerIp>
      <InstanceId>xxx</InstanceId>
      <InstanceContainerStatus>Running</InstanceContainerStatus>
      <GroupId>00000000-0000-0000-0000-000000000000</GroupId>
    </Instances>
    <TotalSize>2</TotalSize>
    <CurrentPage>1</CurrentPage>
  </Data>
  <Message>success</Message>
  <RequestId>00000000-0000-0000-0000-000000000000</RequestId>
  <TraceId>00000000000000000000000000000000</TraceId>
  <ErrorCode>success</ErrorCode>
  <Success>>true</Success>
  <Code>200</Code>
</DescribeApplicationInstancesResponse>
```

### JSON 格式

```
{
  "Data": {
    "PageSize": 1,
    "Instances": [
      {
        "CreateTimeStamp": 1558442609000,
        "InstanceContainerIp": "192.168.1.1",
        "InstanceId": "xxx",
        "GroupId": "00000000-0000-0000-0000-000000000000",
        "InstanceContainerStatus": "Running"
      }
    ],
    "CurrentPage": 1,
    "TotalSize": 2
  },
  "Message": "success",
  "RequestId": "00000000-0000-0000-0000-000000000000",
  "TraceId": "00000000000000000000000000000000",
  "Success": true,
  "ErrorCode": "success",
  "Code": 200
}
```

## 错误码

访问[错误中心](#)查看更多错误码。

## 7.8 DescribeApplicationGroups

获取应用实例分组

调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

请求头

该接口使用公共请求头，无特殊请求头。请参见公共请求参数文档。

请求语法

```
GET /pop/v1/sam/app/describeApplicationGroups HTTP/1.1
```

请求参数

名称	类型	是否必选	示例值	描述
AppId	String	是	00000000-0000-0000-0000-000000000000	应用ID
CurrentPage	Integer	否	1	页码
PageSize	Integer	否	10	页面大小

返回数据

名称	类型	示例值	描述
Code	String	200	接口状态或 POP 错误码
Data			应用分组信息
EdasContainerVersion	String	3.5.3	部署时所使用的 EdasContainer 版本。如3.5.3
GroupId	String	00000000-0000-0000-0000-000000000000	分组ID
GroupName	String	_DEFAULT_GROUP	分组名称。 · _DEFAULT_GROUP：默认分组

名称	类型	示例值	描述
GroupType	Integer	0	分组类型
ImageUrl	String	registry-vpc. cn-hangzhou. aliyuncs.com /demo/nginx: latest	部署时设置的 ImageUrl
Jdk	String	Open JDK 8	部署时设置的 JDK
PackageType	String	Image	部署类型。 · Image: 采用 ImageUrl 进行部署
PackageUrl	String	registry-vpc. cn-hangzhou. aliyuncs.com /demo/nginx: latest	部署时上传/设置的 WAR/JAR 包地址
PackageVersion	String	1.0.0	部署时设置的软件版本。若为 ImageUrl 部署, 自动生成
Replicas	Integer	10	所有实例数
RunningInstances	Integer	1	运行中的实例数
WebContainer	String	Apache Tomcat 7	部署时设置的 WebContainer
ErrorCode	String	success	错误码
Message	String	success	附加信息
RequestId	String	00000000-0000 -0000-0000- 000000000000	请求ID
Success	Boolean	true	是否成功
TraceId	String	0000000000 0000000000 0000000000	调用链ID

## 示例

### 请求示例

```
GET /pop/v1/sam/app/describeApplicationGroups HTTP/1.1
公共请求头
{
  "AppId": ""
}
```

### 正常返回示例

#### XML 格式

```
<DescribeApplicationGroupsResponse>
  <Data>
    <GroupType>0</GroupType>
    <RunningInstances>2</RunningInstances>
    <PackageType>Image</PackageType>
    <GroupName>_DEFAULT_GROUP</GroupName>
    <PackageVersion>2019-04-26 15:56:34.115</PackageVersion>
    <ImageUrl>registry-vpc.cn-hangzhou.aliyuncs.com/demo/nginx:
latest</ImageUrl>
    <Replicas>2</Replicas>
    <GroupId>00000000-0000-0000-0000-000000000000</GroupId>
  </Data>
  <Message>success</Message>
  <RequestId>00000000-0000-0000-0000-000000000000</RequestId>
  <TraceId>00000000000000000000000000000000</TraceId>
  <ErrorCode>success</ErrorCode>
  <Success>>true</Success>
  <Code>200</Code>
</DescribeApplicationGroupsResponse>
```

#### JSON 格式

```
{
  "Data": [
    {
      "GroupType": 0,
      "PackageType": "Image",
      "RunningInstances": 2,
      "GroupName": "_DEFAULT_GROUP",
      "PackageVersion": "2019-04-26 15:56:34.115",
      "ImageUrl": "registry-vpc.cn-hangzhou.aliyuncs.com/demo/nginx:latest",
      "Replicas": 2,
      "GroupId": "00000000-0000-0000-0000-000000000000"
    }
  ],
  "Message": "success",
  "RequestId": "00000000-0000-0000-0000-000000000000",
  "TraceId": "00000000000000000000000000000000",
  "Success": true,
  "ErrorCode": "success",
  "Code": 200
}
```

```
}
```

### 错误码

访问[错误中心](#)查看更多错误码。

## 7.9 ListApplications

获取应用列表

### 调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

### 请求头

该接口使用公共请求头，无特殊请求头。请参见公共请求参数文档。

### 请求语法

### 请求参数

名称	类型	是否必选	示例值	描述
CurrentPage	Integer	是	1	当前页数
AppName	String	否	demo-app	应用名
PageSize	Integer	否	20	页大小

### 返回数据

名称	类型	示例值	描述
Code	String	200	接口状态
Data			应用列表
Applications			应用列表
AppDeletingStatus	Boolean	false	是否正在删除



名称	类型	示例值	描述
AppId	String	f7730764-d88f-4b9a-8d8e-cd8efbfee1f0	应用ID
AppName	String	demo-app	应用名称
Instances	Integer	2	应用节点个数
RegionId	String	cn-beijing	Region ID
RunningInstances	Integer	2	运行中的实例个数
CurrentPage	Integer	1	当前页数
PageSize	Integer	20	页大小
TotalSize	Integer	2	应用总个数
ErrorCode	String	success	错误码
Message	String	success	附加信息
RequestId	String	B4D805CA-926D-41B1-8E63-7AD0C1EDBE8A	请求ID
Success	Boolean	true	是否成功标识

## 示例

### 请求示例

```
http(s)://[Endpoint]/?<公共请求参数>
```

### 正常返回示例

#### JSON 格式

```
{
  "Data":{
    "PageSize":10,
    "Applications":[
      {
        "RunningInstances":0,
```

```

    "RegionId":"cn-beijing",
    "AppId":"f7730764-d88f-4b9a-8d8e-cd8efbfee1f0",
    "Instances":2,
    "AppName":"xxxx",
    "AppDeletingStatus":false
  },
  {
    "RunningInstances":0,
    "RegionId":"cn-beijing",
    "AppId":"a4a60839-3464-4d7b-8b33-9e8697b7b156",
    "Instances":1,
    "AppName":"xxxxx",
    "AppDeletingStatus":false
  }
],
"TotalSize":2,
"CurrentPage":1
},
"Message":"success",
"RequestId":"B4D805CA-926D-41B1-8E63-7AD0C1EDBE8A",
"Success":true,
"ErrorCode":"success",
"Code":200
}

```

### 错误码

访问[错误中心](#)查看更多错误码。

## 7.10 QueryResourceStatics

获取应用的资源使用量

### 调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

### 请求头

该接口使用公共请求头，无特殊请求头。请参见公共请求参数文档。

### 请求语法

### 请求参数

名称	类型	是否必选	示例值	描述
AppId	String	是	7171a6ca-d1cd-4928-8642-7d5cfe69a26c	应用ID

## 返回数据

名称	类型	示例值	描述
Code	String	200	接口状态
Data			资源使用信息
RealTimeRes			实时资源使用量
Cpu	Integer	13	CPU使用量, 单位 Core · min
Memory	Integer	26	内存使用量, 单位 GiB · min
Summary			当月资源使用信息
Cpu	Integer	3354	CPU使用量, 单位 Core · min
Memory	Integer	6708	内存使用量, 单位 GiB · min
ErrorCode	String	success	错误码
Message	String	success	附加信息
RequestId	String	7CCF7092-72CA-4431-90D6-C7D9875267AA	请求ID
Success	Boolean	true	是否成功标识
TraceId	String	ac1a08a015623098794277264e20c8	调用ID

## 示例

## 请求示例

```


```

```
http(s)://[Endpoint]/?<公共请求参数>
```

### 正常返回示例

#### JSON 格式

```
{
  "Data":{
    "RealTimeRes":{
      "Cpu":13,
      "Memory":26
    },
    "Summary":{
      "Cpu":3354,
      "Memory":6708
    }
  },
  "Message":"success",
  "RequestId":"7CCF7092-72CA-4431-90D6-C7D9875267AA",
  "TraceId":"ac1a08a015623098794277264e20c8",
  "Success":true,
  "ErrorCode":"success",
  "Code":200
}
```

### 错误码

访问[错误中心](#)查看更多错误码。

## 7.11 ListLogConfigs

获取应用日志列表

### 调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

### 请求头

该接口使用公共请求头，无特殊请求头。请参见公共请求参数文档。

### 请求语法

```
GET /pop/v1/sam/log/listLogConfigs HTTP/1.1
```

### 请求参数

名称	类型	是否必选	示例值	描述
AppId	String	是	56f77b65-788d-442a-9885-7f20d91f3672	应用ID

名称	类型	是否必选	示例值	描述
CurrentPage	Integer	是	1	当前页
PageSize	Integer	是	10	页大小

## 返回数据

名称	类型	示例值	描述
Code	String	200	接口返回状态
Data			文件日志信息
CurrentPage	Integer	1	当前页
LogConfigs			日志配置
ConfigName	String	sae-1f240907a6 faf58c653f 09e81b7e316e	日志服务配置名称
CreateTime	String	2019-08-29 17:18 :00	日志配置创建时间
LogDir	String	/root/logs/hsf/ hsf.log	日志文件路径
LogType	String	file_log	日志类型，当前只支持文件日志
RegionId	String	cn-beijing	Region ID
SlsLogStore	String	sae-1f240907a6 faf58c653f 09e81b7e316e	日志服务日志存储名
SlsProject	String	sae-56f77b65- 788d-442a-9885- 7f20d91f3672	日志服务Project
StoreType	String	sls	日志服务存储类型
PageSize	Integer	10	页大小
TotalSize	Integer	1	总条数

名称	类型	示例值	描述
ErrorCode	String	success	错误码
Message	String	success	附件信息
RequestId	String	ac1d5e2c15 6715812524 13581d37bd	请求ID
Success	Boolean	true	是否成功
TraceId	String	ac1d5e2c15 6715812524 13581d37bd	调用链ID

## 示例

### 请求示例

```
GET /pop/v1/sam/log/listLogConfigs HTTP/1.1
公共请求头
{
  "AppId": ""
  "CurrentPage":
  "PageSize":
}
```

### 正常返回示例

#### JSON 格式

```
{
  "successResponse":true,
  "requestId":"ac1d5e2c15671581252413581d37bd",
  "data":{
    "Data":{
      "PageSize":10,
      "TotalSize":1,
      "CurrentPage":1,
      "LogConfigs":[
        {
          "SlsLogStore":"sae-1f240907a6faf58c653f09e81b7e316e",
          "LogDir":"/root/logs/hsf/hsf.log",
          "RegionId":"cn-beijing",
          "CreateTime":"2019-08-29 17:18:00",
          "ConfigName":"sae-1f240907a6faf58c653f09e81b7e316e",
          "StoreType":"sls",
          "LogType":"file_log",
          "SlsProject":"sae-56f77b65-788d-442a-9885-7f20d91f3672"
        }
      ]
    }
  },
  "Message":"success",
  "TraceId":"ac1d5e2c15671581252413581d37bd",
```

```
"Success":true,  
"ErrorCode":"success",  
"Code":200  
},  
"code":"200"  
}
```

## 错误码

访问[错误中心](#)查看更多错误码。

## 8 应用生命周期

### 8.1 DescribeApplicationStatus

获取应用的状态信息

调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

请求头

该接口使用公共请求头，无特殊请求头。请参见公共请求参数文档。

请求语法

```
GET /pop/v1/sam/app/describeApplicationStatus HTTP/1.1
```

请求参数

名称	类型	是否必选	示例值	描述
AppId	String	是	0099b7be-5f5b-4512-a7fc-56049ef1aa6b	目标应用ID

返回数据

名称	类型	示例值	描述
Code	String	200	接口状态或 POP 错误码
Data			返回结果
AppId	String	0099b7be-5f5b-4512-a7fc-56049ef1aa6b	当前应用ID
ArmsAdvancedEnabled	String	false	是否开通ARMS高级版



名称	类型	示例值	描述
ArmsApmInfo	String	{"appId": "0099b7be-5f5b-4512-a7fc-56049ef1aa6b", "licenseKey": "d5cgmt5pu0@7303f55292a2d5f"}	该应用在ARMS侧产品元数据信息
CreateTime	String	1563373372746	应用创建时间
CurrentStatus	String	RUNNING	应用当前状态
LastChangeOrderId	String	1ccc2339-fc19-49aa-bda0-1e7b84977515	上一次执行的发布单ID，未执行过或发布单信息过期则为空
LastChangeOrderRunning	Boolean	false	上一次发布单是否处于执行中
LastChangeOrderStatus	String	SUCCESS	上一次发布单状态
RunningInstances	Integer	1	当前应用运行中的实例数
ErrorCode	String	success	错误码
Message	String	success	附加信息
RequestId	String	00000000-0000-0000-0000-000000000000	请求ID
Success	Boolean	true	是否成功
TraceId	String	0000000000 0000000000 0000000000	调用链ID

## 示例

### 请求示例

```
GET /pop/v1/sam/app/describeApplicationStatus HTTP/1.1
```

**公共请求头**

```
{
  "AppId": "0099b7be-5f5b-4512-a7fc-56049ef1aa6b"
}
```

**正常返回示例****XML 格式**

```
<DescribeApplicationStatusResponse>
  <Code>200</Code>
  <Data>
    <AppId>0099b7be-5f5b-4512-a7fc-56049ef1aa6b</AppId>
    <ArmsAdvancedEnabled>>false</ArmsAdvancedEnabled>
    <ArmsApmInfo>
      <appId>0099b7be-5f5b-4512-a7fc-56049ef1aa6b</appId>
      <licenseKey>d5cgdt5pu0@7303f55292a2d5f</licenseKey>
    </ArmsApmInfo>
    <CreateTime>1563373372746</CreateTime>
    <CurrentStatus>RUNNING</CurrentStatus>
    <LastChangeOrderId>1ccc2339-fc19-49aa-bda0-1e7b84977515</
LastChangeOrderId>
    <LastChangeOrderRunning>>false</LastChangeOrderRunning>
    <LastChangeOrderStatus>SUCCESS</LastChangeOrderStatus>
    <RunningInstances>1</RunningInstances>
  </Data>
  <ErrorCode>success</ErrorCode>
  <Message>success</Message>
  <RequestId>00000000-0000-0000-0000-000000000000</RequestId>
  <Success>>true</Success>
  <TraceId>00000000000000000000000000000000</TraceId>
</DescribeApplicationStatusResponse>
```

**JSON 格式**

```
{
  "Data": {
    "LastChangeOrderStatus": "SUCCESS",
    "RunningInstances": 1,
    "CreateTime": "1563373372746",
    "ArmsAdvancedEnabled": false,
    "AppId": "0099b7be-5f5b-4512-a7fc-56049ef1aa6b",
    "CurrentStatus": "RUNNING",
    "LastChangeOrderId": "1ccc2339-fc19-49aa-bda0-1e7b84977515",
    "ArmsApmInfo": {
      "appId": "0099b7be-5f5b-4512-a7fc-56049ef1aa6b",
      "licenseKey": "d5cgdt5pu0@7303f55292a2d5f"
    },
    "LastChangeOrderRunning": false
  },
  "Message": "success",
  "RequestId": "00000000-0000-0000-0000-000000000000",
  "TraceId": "00000000000000000000000000000000",
  "Success": true,
  "ErrorCode": "success",
  "Code": 200
}
```

```
}

```

### 错误码

HttpCode	错误码	错误信息	描述
400	InvalidApplication.NotFound	The current application does not exist.	找不到当前应用
400	InvalidParameter.NotEmpty	You must specify the parameter %s.	不合法的参数: %s 不能为空

访问[错误中心](#)查看更多错误码。

## 8.2 DeployApplication

### 部署应用

#### 调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

#### 请求头

该接口使用公共请求头，无特殊请求头。请参见公共请求参数文档。

#### 请求语法

```
POST /pop/v1/sam/app/deployApplication HTTP/1.1
```

#### 请求参数

名称	类型	是否必选	示例值	描述
AppId	String	否	017f39b8-dfa4-4e16-a84b-1dcee4b17106	需要部署的应用ID
BatchWaitTime	Integer	否	10	分批等待时间。单位秒。
Command	String	否	sleep	镜像启动命令。该命令必须为容器内存在的可执行的对象。例如: sleep。设置该命令将导致镜像原本的启动命令失效。

名称	类型	是否必选	示例值	描述
CommandArgs	String	否	1d	镜像启动命令参数。上述启动命令所需参数。例如： <code>"1d"</code>
CustomHostAlias	String	否	<code>[{"hostName":"samplehost","ip":"127.0.0.1"}]</code>	容器内自定义host映射。例如： <code>{"hostName":"samplehost","ip":"127.0.0.1"}</code>
EdasContainerVersion	String	否	3.5.3	EDAS pandora应用使用的运行环境
Envs	String	否	<code>[{"name":"envtmp","value":"0"}]</code>	容器环境变量参数。例如： <code>{"name":"envtmp","value":"0"}</code>
ImageUrl	String	否	registry.cn-hangzhou.aliyuncs.com/sae_test/ali_sae_test:0.0.1	镜像地址。只有Image类型应用可以配置镜像地址。
JarStartArgs	String	否	custom-args	Jar包启动应用参数。应用默认启动命令： <code>\$JAVA_HOME/bin/java \$JarStartOptions -jar \$CATALINA_OPTS "\$package_path" \$JarStartArgs</code>
JarStartOptions	String	否	custom-option	Jar包启动应用选项。应用默认启动命令： <code>\$JAVA_HOME/bin/java \$JarStartOptions -jar \$CATALINA_OPTS "\$package_path" \$JarStartArgs</code>
Jdk	String	否	Open JDK 8	部署的包依赖的JDK版本。镜像不支持。

名称	类型	是否必选	示例值	描述
Liveness	String	否	<code>{"exec":{"command":["sleep","5s"],"initialDelaySeconds":10,"timeoutSeconds":11}</code>	容器健康检查，健康检查失败的容器将被杀死并恢复。目前仅支持容器内下发命令的方式。 <code>{"exec":{"command":"sleep","5s"},"initialDelaySeconds":10,"timeoutSeconds":11}</code>
MinReadyInstances	Integer	否	1	最小可用实例数。任何变更都尽量不会违背这个设置，保证业务稳定性。
PackageUrl	String	否	<code>http://myoss.oss-cn-hangzhou.aliyuncs.com/my-buc/2019-06-30/sae-test.jar</code>	部署包地址。只有FatJar或War类型应用可以配置部署包地址。
PackageVersion	String	否	1.0.1	部署的包的版本号，War FatJar类型必填。请自定义它的含义。
Readiness	String	否	<code>{"exec":{"command":["sleep","6s"],"initialDelaySeconds":15,"timeoutSeconds":12}</code>	应用启动状态检查，多次健康检查失败的容器将被杀死并重启。不通过健康检查的容器将不会有SLB流量进入。例如： <code>{"exec":{"command":"sleep","6s"},"initialDelaySeconds":15,"timeoutSeconds":12}</code>
UpdateStrategy	String	否	<code>{}</code>	部署策略
WebContainer	String	否	<code>apache-tomcat-7.0.91</code>	部署的包依赖的tomcat版本。镜像不支持。

## 返回数据

名称	类型	示例值	描述
Code	String	200	接口状态或 POP 错误码

名称	类型	示例值	描述
Data			返回结果
AppId	String	017f39b8-dfa4-4e16-a84b-1dcee4b17106	应用ID。
ChangeOrderId	String	01db03d3-3ee9-48b3-b3d0-dfce2d88ab31	返回的发布单ID，用于查询任务执行状态。
ErrorCode	String	success	错误码
Message	String	success	附加信息
RequestId	String	00000000-0000-0000-0000-000000000000	请求ID
Success	Boolean	true	是否成功
TraceId	String	00000000-0000-0000-0000-000000000000	调用链ID

## 示例

### 请求示例

```
POST /pop/v1/sam/app/deployApplication HTTP/1.1
公共请求头
```

### 正常返回示例

#### XML 格式

```
<DeployApplication>
  <Data>
    <ChangeOrderId>01db03d3-3ee9-48b3-b3d0-dfce2d88ab31</ChangeOrderId>
    <AppId>017f39b8-dfa4-4e16-a84b-1dcee4b17106</AppId>
  </Data>
  <Message>success</Message>
  <RequestId>00000000-0000-0000-0000-000000000000</RequestId>
  <TraceId>00000000000000000000000000000000</TraceId>
  <Success>>true</Success>
  <ErrorCode>success</ErrorCode>
  <Code>200</Code>
```

```
</DeployApplication>
```

### JSON 格式

```
{
  "Data":{
    "AppId":"017f39b8-dfa4-4e16-a84b-1dcee4b17106",
    "ChangeOrderId":"01db03d3-3ee9-48b3-b3d0-dfce2d88ab31"
  },
  "Message":"success",
  "RequestId":"00000000-0000-0000-0000-000000000000",
  "TraceId":"00000000000000000000000000000000",
  "ErrorCode":"success",
  "Success":true,
  "Code":200
}
```

### 错误码

HttpCode	错误码	错误信息	描述
400	Application.ChangerOrderRunning	An application change process is in progress. Please try again later.	应用有变更流程正在执行，请稍后重试。
400	Application.InvalidStatus	The application status is abnormal. Please try again later.	应用状态异常，请稍后重试。
400	Application.MissingJdk	Your application must at least contain a JDK component.	应用必须至少包含JDK组件
400	InvalidApplication.NotFound	The current application does not exist.	找不到当前应用
400	InvalidComponent.NotFound	The current component (such as JDK, Tomcat, or EDASWebContainer) does not exist.	找不到当前组件（JDK、Tomcat、EDASWebContainer等）
400	InvalidHostnameIp.Invalid	The hostname and/or IP is invalid: Hostname [%s], IP [%s].	主机名和/或IP不合法：主机名[%s]，IP[%s]。
400	InvalidInstanceSpecification.Unsupported	The instance specification is not supported: CPU [%s], memory [%s].	不支持的实例规格。CPU[%s]，Memory[%s]。

HttpCode	错误码	错误信息	描述
404	InvalidNamespaceId.NotFound	The specified NamespaceId does not exist.	指定的NamespaceId不存在。
400	InvalidPackageType.NotFound	The package type must be War, FatJar, or Image.	包类型必须为War、Fatjar或Image
400	InvalidParameter.FileName	The application deployment package name is invalid. This name can contain only alphanumeric characters, hyphens (-), and underscores (_). In addition, you can upload JAR files only if the selected deployment version supports JAR file. Otherwise, upload WAR files only.	应用部署程序包名称无效。名称只能包含字母、数字和特殊字符“-”“_”。另外，仅当选择的部署版本支持JAR部署时方可上传JAR包，否则只能上传WAR包。
400	InvalidParameter.NotEmpty	You must specify the parameter %s.	不合法的参数：%s 不能为空
400	InvalidParameter.Obviously	The specified parameter is invalid %s}.	不合法的参数{%s}
400	InvalidParameter.WithMessage	The parameter is invalid %s}: %s	不合法的参数{%s}: %s
400	JarApplication.MissingJdk	A FatJar application must contain JDK.	FatJAR类型应用必须包含JDK
400	NoComputeResourceQuota.Exceed	Your compute resource is insufficient. Please submit a ticket to raise the quota.	计算资源不足，请提交工单增加计算资源额度。
400	PandoraApplication.MissingJdk	The Pandora application is missing a JDK component.	pandora应用缺少JDK组件
400	PandoraApplication.OnlyJdk	A Pandora application only requires JDK component.	pandora应用只需要JDK组件



HttpCode	错误码	错误信息	描述
400	WarApplication.MissingJdkWebcontainer	A War application must contain JDK and Tomcat.	War类型应用必须包含JDK和Tomcat

访问[错误中心](#)查看更多错误码。

## 8.3 CreateApplication

创建一个SAE应用

调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

请求头

该接口使用公共请求头，无特殊请求头。请参见公共请求参数文档。

请求语法

```
POST /pop/v1/sam/app/createApplication HTTP/1.1
```

请求参数

名称	类型	是否必选	示例值	描述
AppName	String	是	test-app	应用名称。允许数字，字母以及中划线组合。必须字母打头，最大长度36个字符
NamespaceId	String	是	cn-beijing:test	EDAS命名空间对应ID。只支持名称为小写字母加中划线的命名空间，必须以字母开头。命名空间可从DescribeNamespaceList接口获取。
PackageType	String	是	FatJar	应用包类型。支持FatJar、War、Image。
Replicas	Integer	是	1	初始实例数。

名称	类型	是否必选	示例值	描述
AppDescription	String	否	This is a test description.	应用描述信息。不超过1024个字符
Command	String	否	sleep	镜像启动命令。该命令必须为容器内存在的可执行的对象。例如: sleep。设置该命令将导致镜像原本的启动命令失效。
CommandArgs	String	否	1d	镜像启动命令参数。上述启动命令所需参数。例如: <code>"1d"</code>
Cpu	Integer	否	1000	每个实例所需的CPU，单位为毫核，不能为0。目前仅支持固定规格的实例类型。
CustomHostAlias	String	否	<code>[{"hostName":"samplehost","ip":"127.0.0.1"}]</code>	容器内自定义host映射。例如: <code>[{"hostName":"samplehost","ip":"127.0.0.1"}]</code>
Deploy	Boolean	否	true	是否立即部署生效。默认为否。
EdasContainerVersion	String	否	3.5.3	EDAS pandora应用使用的运行环境
Envs	String	否	<code>[{"name":"envtmp","value":"0"}]</code>	容器环境变量参数。例如: <code>[{"name":"envtmp","value":"0"}]</code>
ImageUrl	String	否	registry.cn-hangzhou.aliyuncs.com/sae_test/ali_sae_test:0.0.1	镜像地址。只有Image类型应用可以配置镜像地址。

名称	类型	是否必选	示例值	描述
JarStartArgs	String	否	custom-args	Jar包启动应用参数。应用默认启动命令: \$JAVA_HOME/bin/java \$JarStartOptions -jar \$CATALINA_OPTS "\$package_path" \$JarStartArgs
JarStartOptions	String	否	custom-option	Jar包启动应用选项。应用默认启动命令: \$JAVA_HOME/bin/java \$JarStartOptions -jar \$CATALINA_OPTS "\$package_path" \$JarStartArgs
Jdk	String	否	Open JDK 8	部署的包依赖的JDK版本。镜像不支持。
Liveness	String	否	<code>{"exec":{"command":["sleep","5s"],"initialDelaySeconds":10,"timeoutSeconds":11}</code>	容器健康检查, 健康检查失败的容器将被杀死并恢复。目前仅支持容器内下发命令的方式。{"exec":{"command":"sleep","5s"},"initialDelaySeconds":10,"timeoutSeconds":11}
Memory	Integer	否	1024	每个实例所需的内存, 单位为MB, 不能为0。目前仅支持固定规格的实例类型。
PackageUrl	String	否	http://myoss.oss-cn-hangzhou.aliyuncs.com/my-buc/2019-06-30/sae-test.jar	部署包地址。只有FatJar或War类型应用可以配置部署包地址。
PackageVersion	String	否	1.0.0	部署的包的版本号, War FatJar类型必填。请自定义它的含义。

名称	类型	是否必选	示例值	描述
Readiness	String	否	<code>{"exec":{"command":["sleep","6s"],"initialDelaySeconds":15,"timeoutSeconds":12}</code>	应用启动状态检查，多次健康检查失败的容器将被杀死并重启。不通过健康检查的容器将不会有SLB流量进入。例如： <code>{"exec":{"command":["sleep","6s"],"initialDelaySeconds":15,"timeoutSeconds":12}</code>
VSwitchId	String	否	vsw-XXXXXXXXXX XXXXXXXXXX	应用实例弹性网卡所在的虚拟交换机。该交换机必须位于上述VPC内。该交换机与EDAS命名空间同样存在绑定关系。不填则为命名空间绑定的VSwitchId。
VpcId	String	否	vpc-XXXXXXXXXX XXXXXXXXXX	EDAS命名空间对应的VPC。在serverless中，一个命名空间只能对应一个VPC，且不能修改。第一次在命名空间内创建Serverless应用将形成绑定关系。多个命名空间可以对应一个VPC。不填则为命名空间绑定的VpcId。
WebContainer	String	否	apache-tomcat-7.0.91	部署的包依赖的tomcat版本。镜像不支持。

## 返回数据

名称	类型	示例值	描述
Code	String	200	接口状态或 POP 错误码
Data			返回结果
AppId	String	017f39b8-dfa4-4e16-a84b-1dcee4b17106	创建成功的应用ID。

名称	类型	示例值	描述
ChangeOrderId	String	01db03d3-3ee9-48b3-b3d0-dfce2d88ab31	返回的发布单ID，用于查询任务执行状态。
ErrorCode	String	success	错误码
Message	String	success	附加信息
RequestId	String	00000000-0000-0000-0000-000000000000	请求ID
Success	Boolean	true	是否成功
TraceId	String	00000000-0000-0000-000000000000	调用链ID

## 示例

### 请求示例

```
POST /pop/v1/sam/app/createApplication HTTP/1.1
公共请求头
{
  "AppName": "test-app"
  "NamespaceId": "cn-beijing:test"
  "PackageType": "FatJar"
  "Replicas": "1"
}
```

### 正常返回示例

#### XML 格式

```
<CreateApplication>
  <Data>
    <ChangeOrderId>01db03d3-3ee9-48b3-b3d0-dfce2d88ab31</ChangeOrderId>
    <AppId>017f39b8-dfa4-4e16-a84b-1dcee4b17106</AppId>
  </Data>
  <Message>success</Message>
  <RequestId>00000000-0000-0000-0000-000000000000</RequestId>
  <TraceId>00000000000000000000000000000000</TraceId>
  <Success>true</Success>
  <ErrorCode>success</ErrorCode>
  <Code>200</Code>
```

```
</CreateApplication>
```

### JSON 格式

```
{
  "Data":{
    "AppId":"017f39b8-dfa4-4e16-a84b-1dcee4b17106",
    "ChangeOrderId":"01db03d3-3ee9-48b3-b3d0-dfce2d88ab31"
  },
  "Message":"success",
  "RequestId":"00000000-0000-0000-0000-000000000000",
  "TraceId":"00000000000000000000000000000000",
  "ErrorCode":"success",
  "Success":true,
  "Code":200
}
```

### 错误码

HttpCode	错误码	错误信息	描述
400	Application.MissingJdk	Your application must at least contain a JDK component.	应用必须至少包含JDK组件
404	InvalidNamespaceId.NotFound	The specified NamespaceId does not exist.	指定的NamespaceId不存在。
400	InvalidPackageType.NotFound	The package type must be War, FatJar, or Image.	包类型必须为War、Fatjar或Image
400	InvalidParameter.FileName	The application deployment package name is invalid. This name can contain only alphanumeric characters, hyphens (-), and underscores (_). In addition, you can upload JAR files only if the selected deployment version supports JAR file. Otherwise, upload WAR files only.	应用部署程序包名称无效。名称只能包含字母、数字和特殊字符“-”“_”。另外，仅当选择的部署版本支持JAR部署时方可上传JAR包，否则只能上传WAR包。
400	InvalidParameter.NotEmpty	You must specify the parameter %s.	不合法的参数：%s 不能为空

HttpCode	错误码	错误信息	描述
400	InvalidParameter. Obviously	The specified parameter is invalid {%s}.	不合法的参数{%s}
400	JarApplication. MissingJdk	A FatJar application must contain JDK.	FatJAR类型应用必须包含JDK
400	NoAvailableCluster. NotFound	No clusters are available for the current region.	当前地域没有可用集群
400	NoComputeResourceQuota. Exceed	Your compute resource is insufficient. Please submit a ticket to raise the quota.	计算资源不足，请提交工单增加计算资源额度。
400	PandoraApplication. MissingJdk	The Pandora application is missing a JDK component.	pandora应用缺少JDK组件
400	PandoraApplication. OnlyJdk	A Pandora application only requires JDK component.	pandora应用只需要JDK组件
400	InvalidParameter. WithMessage	The parameter is invalid {%s}: %s	不合法的参数{%s}: %s
400	InvalidComponent. NotFound	The current component (such as JDK, Tomcat, or EDASWebContainer) does not exist.	找不到当前组件 (JDK、Tomcat、EDASWebContainer等)
400	InvalidHostnameIp. Invalid	The hostname and/or IP is invalid: Hostname [%s], IP [%s].	主机名和/或IP不合法: 主机名[%s], IP[%s]。
400	InvalidInstanceSpecification. Unsupported	The instance specification is not supported: CPU [%s], memory [%s].	不支持的实例规格。CPU[%s], Memory[%s]。
400	InvalidServerlessRegion. Unsupported	The current region is not supported: %s	不支持当前地域: %s
404	InvalidVpcId. NotFound	The specified VpcId does not exist.	指定的VpcId不存在。

HttpCode	错误码	错误信息	描述
400	WarApplication.MissingJdkWebcontainer	A War application must contain JDK and Tomcat.	War类型应用必须包含JDK和Tomcat
400	InvalidNamespace.WithUppercase	This namespace does not support creating SAE apps because it contains uppercase letters.	命名空间不支持创建SAE应用，因为它带有大写字母。

访问[错误中心](#)查看更多错误码。

## 8.4 DeleteApplication

删除应用

调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

请求头

该接口使用公共请求头，无特殊请求头。请参见公共请求参数文档。

请求语法

```
DELETE /pop/v1/sam/app/deleteApplication HTTP/1.1
```

请求参数

名称	类型	是否必选	示例值	描述
AppId	String	是	017f39b8-dfa4-4e16-a84b-1dcee4b17106	要删除的应用ID

返回数据

名称	类型	示例值	描述
Code	String	200	接口状态或 POP 错误码
Data			返回结果



名称	类型	示例值	描述
ChangeOrderId	String	01db03d3-3ee9-48b3-b3d0-dfce2d88ab31	返回的发布单ID，用于查询任务执行状态。
ErrorCode	String	success	错误码
Message	String	success	附加信息
RequestId	String	00000000-0000-0000-0000-000000000000	请求ID
Success	Boolean	true	是否成功
TraceId	String	00000000-0000-0000-0000-000000000000	调用链ID

## 示例

### 请求示例

```
DELETE /pop/v1/sam/app/deleteApplication HTTP/1.1
公共请求头
{
  "AppId": "017f39b8-dfa4-4e16-a84b-1dcee4b17106"
}
```

### 正常返回示例

#### XML 格式

```
<DeleteApplication>
  <Data>
    <ChangeOrderId>01db03d3-3ee9-48b3-b3d0-dfce2d88ab31</ChangeOrderId>
  </Data>
  <Message>success</Message>
  <RequestId>00000000-0000-0000-0000-000000000000</RequestId>
  <TraceId>00000000000000000000000000000000</TraceId>
  <Success>>true</Success>
  <ErrorCode>success</ErrorCode>
  <Code>200</Code>
</DeleteApplication>
```

#### JSON 格式

```
{
  "Data": {
    "ChangeOrderId": "01db03d3-3ee9-48b3-b3d0-dfce2d88ab31"
  },
}
```

```

"Message": "success",
"RequestId": "000000000-0000-0000-0000-0000000000000",
"TraceId": "00000000000000000000000000000000",
"ErrorCode": "success",
"Success": true,
"Code": 200
}

```

## 错误码

HttpCode	错误码	错误信息	描述
400	Application.ChangerOrderRunning	An application change process is in progress. Please try again later.	应用有变更流程正在执行，请稍后重试。
400	InvalidApplication.NotFound	The current application does not exist.	找不到当前应用
400	InvalidParameter.NotEmpty	You must specify the parameter %s.	不合法的参数： %s 不能为空
400	InvalidParameter.Obviously	The specified parameter is invalid %s}.	不合法的参数{ %s}
400	InvalidParameter.WithMessage	The parameter is invalid %s}: %s	不合法的参数{ %s}: %s

访问[错误中心](#)查看更多错误码。

## 8.5 StopApplication

### 停止应用

#### 调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

#### 请求头

该接口使用公共请求头，无特殊请求头。请参见公共请求参数文档。

## 请求语法

```
PUT /pop/v1/sam/app/stopApplication HTTP/1.1
```

## 请求参数

名称	类型	是否必选	示例值	描述
AppId	String	是	0099b7be-5f5b-4512-a7fc-56049ef1aa6b	目标应用ID

## 返回数据

名称	类型	示例值	描述
Code	String	200	接口状态或 POP 错误码
Data			返回结果
ChangeOrderId	String	4a815998-b468-4bea-b7d8-59f52a4421f5	发布单ID
ErrorCode	String	success	错误码
Message	String	success	附加信息
RequestId	String	00000000-0000-0000-0000-000000000000	请求ID
Success	Boolean	true	是否成功
TraceId	String	0000000000 0000000000 0000000000	调用链ID

## 示例

## 请求示例

```
PUT /pop/v1/sam/app/stopApplication HTTP/1.1
公共请求头
{
  "AppId": "0099b7be-5f5b-4512-a7fc-56049ef1aa6b"
}
```

## 正常返回示例

## XML 格式

```
<StopApplicationResponse>
  <Message>success</Message>
  <RequestId>7BD8F4C7-D84C-4D46-9885-8212997E24D6</RequestId>
  <TraceId>0bc3b6e215637275918588187d221e</TraceId>
  <Data>
    <ChangeOrderId>4a815998-b468-4bea-b7d8-59f52a4421f5</ChangeOrder
    rId>
  </Data>
  <ErrorCode>success</ErrorCode>
  <Code>200</Code>
  <Success>true</Success>
</StopApplicationResponse>
```

## JSON 格式

```
{
  "Data":{
    "ChangeOrderId":"4a815998-b468-4bea-b7d8-59f52a4421f5"
  },
  "Message":"success",
  "RequestId":"7BD8F4C7-D84C-4D46-9885-8212997E24D6",
  "TraceId":"0bc3b6e215637275918588187d221e",
  "Success":true,
  "ErrorCode":"success",
  "Code":200
}
```

## 错误码

HttpCode	错误码	错误信息	描述
400	Application.ChangerOrderRunning	An application change process is in progress. Please try again later.	应用有变更流程正在执行，请稍后重试。
400	Application.InvalidStatus	The application status is abnormal. Please try again later.	应用状态异常，请稍后重试。
400	Application.NotDeployYet	The application has not been deployed. Please deploy it and try again.	应用没有部署，请部署后重试。
400	InvalidApplication.NotFound	The current application does not exist.	找不到当前应用
400	InvalidParameter.NotEmpty	You must specify the parameter %s.	不合法的参数：%s 不能为空

HttpCode	错误码	错误信息	描述
400	InvalidParameter. Obviously	The specified parameter is invalid {%s}.	不合法的参数{%s}
400	InvalidParameter. WithMessage	The parameter is invalid {%s}: %s	不合法的参数{%s}: %s

访问[错误中心](#)查看更多错误码。

## 8.6 RescaleApplicationVertically

改变应用实例规格

调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

请求头

该接口使用公共请求头，无特殊请求头。请参见公共请求参数文档。

请求语法

```
POST /pop/v1/sam/app/rescaleApplicationVertically HTTP/1.1
```

请求参数

名称	类型	是否必选	示例值	描述
AppId	String	是	0099b7be-5f5b-4512-a7fc-56049ef1aa6b	目标应用ID
Cpu	String	是	1000	目标CPU规格，单位毫核。目前只支持有限几种规格。
Memory	String	是	2048	目标memory规格，单位MB。目前只支持有限几种规格。

## 返回数据

名称	类型	示例值	描述
Code	String	200	接口状态或 POP 错误码
Data			返回结果
ChangeOrderId	String	ffd8cd45-2b5f-415d-b4d0-1003e80bf354	发布单ID
ErrorCode	String	success	错误码
Message	String	success	附加信息
RequestId	String	00000000-0000-0000-0000-000000000000	请求ID
Success	Boolean	true	是否成功
TraceId	String	0000000000 0000000000 0000000000	调用链ID

## 示例

## 请求示例

```
POST /pop/v1/sam/app/rescaleApplicationVertically HTTP/1.1
公共请求头
{
  "AppId": "0099b7be-5f5b-4512-a7fc-56049ef1aa6b"
  "Cpu": "1000"
  "Memory": "2048"
}
```

## 正常返回示例

## XML 格式

```
<RescaleApplicationVerticallyResponse>
  <Message>success</Message>
  <RequestId>AB521DBB-FA78-42E6-803F-A862EA4FE908</RequestId>
  <TraceId>0bc3b6f315637273629117900d515f</TraceId>
  <Data>
    <ChangeOrderId>ffd8cd45-2b5f-415d-b4d0-1003e80bf354</ChangeOrderId>
  </Data>
  <ErrorCode>success</ErrorCode>
  <Code>200</Code>
```

```
<Success>true</Success>
</RescaleApplicationVerticallyResponse>
```

### JSON 格式

```
{
  "Data":{
    "ChangeOrderId":"ffd8cd45-2b5f-415d-b4d0-1003e80bf354"
  },
  "Message":"success",
  "RequestId":"AB521DBB-FA78-42E6-803F-A862EA4FE908",
  "TraceId":"0bc3b6f315637273629117900d515f",
  "Success":true,
  "ErrorCode":"success",
  "Code":200
}
```

### 错误码

HttpCode	错误码	错误信息	描述
400	Application.ChangerOrderRunning	An application change process is in progress. Please try again later.	应用有变更流程正在执行，请稍后重试。
400	Application.InvalidStatus	The application status is abnormal. Please try again later.	应用状态异常，请稍后重试。
400	Application.NotDeployYet	The application has not been deployed. Please deploy it and try again.	应用没有部署，请部署后重试。
400	InvalidApplication.NotFound	The current application does not exist.	找不到当前应用
400	InvalidInstanceSpecification.Unsupported	The instance specification is not supported: CPU [%s], memory [%s].	不支持的实例规格。CPU[%s]，Memory[%s]。
400	InvalidParameter.NotEmpty	You must specify the parameter %s.	不合法的参数：%s 不能为空
400	InvalidParameter.Obviously	The specified parameter is invalid {%s}.	不合法的参数{%s}
400	InvalidParameter.WithMessage	The parameter is invalid {%s}: %s	不合法的参数{%s}：%s

HttpCode	错误码	错误信息	描述
400	NoComputeResourceQuota.Exceed	Your compute resource is insufficient. Please submit a ticket to raise the quota.	计算资源不足，请提交工单增加计算资源额度。

访问[错误中心](#)查看更多错误码。

## 8.7 StartApplication

启动应用

调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

请求头

该接口使用公共请求头，无特殊请求头。请参见公共请求参数文档。

请求语法

```
PUT /pop/v1/sam/app/startApplication HTTP/1.1
```

请求参数

名称	类型	是否必选	示例值	描述
AppId	String	是	0099b7be-5f5b-4512-a7fc-56049ef1aa6b	目标应用ID

返回数据

名称	类型	示例值	描述
Code	String	200	接口状态或 POP 错误码
Data			返回结果
ChangeOrderId	String	4a815998-b468-4bea-b7d8-59f52a4421f5	发布单ID
ErrorCode	String	success	错误码



名称	类型	示例值	描述
Message	String	success	附加信息
RequestId	String	00000000-0000-0000-0000-000000000000	请求ID
Success	Boolean	true	是否成功
TraceId	String	0000000000 0000000000 0000000000	调用链ID

## 示例

### 请求示例

```
PUT /pop/v1/sam/app/startApplication HTTP/1.1
公共请求头
{
  "AppId": "0099b7be-5f5b-4512-a7fc-56049ef1aa6b"
}
```

### 正常返回示例

#### XML 格式

```
<StartApplicationResponse>
  <Message>success</Message>
  <RequestId>7BD8F4C7-D84C-4D46-9885-8212997E24D6</RequestId>
  <TraceId>0bc3b6e215637275918588187d221e</TraceId>
  <Data>
    <ChangeOrderId>4a815998-b468-4bea-b7d8-59f52a4421f5</ChangeOrderId>
  </Data>
  <ErrorCode>success</ErrorCode>
  <Code>200</Code>
  <Success>true</Success>
</StartApplicationResponse>
```

#### JSON 格式

```
{
  "Data": {
    "ChangeOrderId": "4a815998-b468-4bea-b7d8-59f52a4421f5"
  },
  "Message": "success",
  "RequestId": "7BD8F4C7-D84C-4D46-9885-8212997E24D6",
  "TraceId": "0bc3b6e215637275918588187d221e",
  "Success": true,
  "ErrorCode": "success",
  "Code": 200
}
```

```
}

```

## 错误码

HttpCode	错误码	错误信息	描述
400	Application.ChangerOrderRunning	An application change process is in progress. Please try again later.	应用有变更流程正在执行, 请稍后重试。
400	Application.InvalidStatus	The application status is abnormal. Please try again later.	应用状态异常, 请稍后重试。
400	Application.NotDeployYet	The application has not been deployed. Please deploy it and try again.	应用没有部署, 请部署后重试。
400	InvalidApplication.NotFound	The current application does not exist.	找不到当前应用
400	InvalidParameter.NotEmpty	You must specify the parameter %s.	不合法的参数: %s 不能为空
400	InvalidParameter.Obviously	The specified parameter is invalid {%s}.	不合法的参数{%s}
400	InvalidParameter.WithMessage	The parameter is invalid {%s}: %s	不合法的参数{%s}: %s
400	NoComputeResourceQuota.Exceed	Your compute resource is insufficient. Please submit a ticket to raise the quota.	计算资源不足, 请提交工单增加计算资源额度。

访问[错误中心](#)查看更多错误码。

## 8.8 ConfirmPipelineBatch

是否开始一下批次

### 调试

您可以在OpenAPI Explorer中直接运行该接口, 免去您计算签名的困扰。运行成功后, OpenAPI Explorer可以自动生成SDK代码示例。

## 请求头

该接口使用公共请求头，无特殊请求头。请参见公共请求参数文档。

## 请求语法

```
GET /pop/v1/sam/changeorder/ConfirmPipelineBatch HTTP/1.1
```

## 请求参数

名称	类型	是否必选	示例值	描述
Confirm	Boolean	是	true	是否开始下一批次
PipelineId	String	是	xxx-xxx-xxx-xxx	批次 ID

## 返回数据

名称	类型	示例值	描述
Code	String	200	接口状态或 POP 错误码
Data			批次信息
PipelineId	String	xxx-xxx-xxx-xxx	批次ID
ErrorCode	String	success	错误码
Message	String	success	附加信息
RequestId	String	0000-0000-0000-0000	请求ID
Success	Boolean	true	是否成功
TraceId	String	0000000000 0000000000 0000000000	调用链ID

## 示例

## 请求示例

```
GET /pop/v1/sam/changeorder/ConfirmPipelineBatch HTTP/1.1
公共请求头
{
  "Confirm": true
  "PipelineId": "xxx-xxx-xxx-xxx"
```

```
}

```

## 正常返回示例

### XML 格式

```
<ConfirmPipelineBatch>
  <Data>
    <PipelineId>xxx-xxx-xxx-xxx</PipelineId>
  </Data>
  <Message>success</Message>
  <RequestId>00000000-0000-0000-0000-000000000000</RequestId>
  <TraceId>00000000000000000000000000000000</TraceId>
  <Success>true</Success>
  <ErrorCode>success</ErrorCode>
  <Code>200</Code>
</ConfirmPipelineBatch>
```

### JSON 格式

```
{
  "Data":{
    "PipelineId":"xxx-xxx-xxx-xxx"
  },
  "Message":"success",
  "RequestId":"00000000-0000-0000-0000-000000000000",
  "TraceId":"00000000000000000000000000000000",
  "ErrorCode":"success",
  "Success":true,
  "Code":200
}
```

## 错误码

HttpCode	错误码	错误信息	描述
400	InvalidParameter. NotEmpty	You must specify the parameter %s.	不合法的参数: %s 不能为空

访问[错误中心](#)查看更多错误码。

## 8.9 ListChangeOrders

获取变更单列表

### 调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

## 请求头

该接口使用公共请求头，无特殊请求头。请参见公共请求参数文档。

## 请求语法

```
GET /pop/v1/sam/changeorder/ListChangeOrders HTTP/1.1
```

## 请求参数

名称	类型	是否必选	示例值	描述
AppId	String	是	145341c-9708-4967-b3ec-24933767fd44	应用ID
CoStatus	String	否	2	0: 准备 1: 执行中 2: 执行成功 3: 执行失败 6: 终止 10: 系统异常 执行失败
CoType	String	否	CoCreateApp	变更单类型
CurrentPage	Integer	否	1	当前分页
Key	String	否	test	模糊查询值
PageSize	Integer	否	20	分页大小

## 返回数据

名称	类型	示例值	描述
Code	String	200	接口状态或 POP 错误码
Data			变更单列表信息
ChangeOrderList			变更单列表信息
AppId	String	164341c-9708-4967-b3ec-24933767fd44	应用ID
BatchCount	Integer	1	分批数
BatchType	String	自动	分批类型

名称	类型	示例值	描述
ChangeOrderId	String	7fa5c0-9ebb-4bb4-b383-1f885447b109	变更单ID
CoType	String	创建应用	变更单类型
CoTypeCode	String	CoCreateApp	变更单类型CODE
CreateTime	String	2019-07-11 15:54:49	创建时间
Description	String	版本: 1.0   镜像名称: nginx	描述信息
FinishTime	String	2019-07-11 20:12:58	结束时间
GroupId	String	c9ecd2-cf6c-46c3-9f20-525de202cf79	分组ID
Source	String	console	变更单操作入口来源
Status	Integer	2	0: 准备 1: 执行中 2: 执行成功 3: 执行失败 6: 终止 10: 系统异常执行失败
CurrentPage	Integer	1	当前分页
PageSize	Integer	20	分页大小
TotalSize	Integer	1	变更单总数
ErrorCode	String	success	错误码
Message	String	success	附加信息
RequestId	String	XXXC03A-E74E-467E-B4E2-63E3F5AACF55	请求ID
Success	Boolean	true	是否成功
TraceId	String	0bc3915638507554994370d5056	调用链ID

## 示例

### 请求示例

```
GET /pop/v1/sam/changeorder/ListChangeOrders HTTP/1.1
公共请求头
{
  "AppId": "145341c-9708-4967-b3ec-24933767fd44"
}
```

### 正常返回示例

### XML 格式

```
<ListChangeOrders>
  <Message>success</Message>
  <RequestId>65E1F-43BA-4D0C-8E61-E4D1337FA9B2</RequestId>
  <TraceId>0bb6f815638568884597879d60be</TraceId>
  <Data>
    <PageSize>20</PageSize>
    <CurrentPage>1</CurrentPage>
    <ChangeOrderList>
      <Status>2</Status>
      <Description>版本: 1.0 | 镜像名称: nginx </Description>
      <CreateTime>2019-07-11 15:54:49</CreateTime>
      <ChangeOrderId>7fa5c0-9ebb-4bb4-b383-1f885447b109</ChangeOrder
rId>
      <BatchType>自动</BatchType>
      <Source>console</Source>
      <GroupId>c9ecd2-cf6c-46c3-9f20-525de202cf79</GroupId>
      <AppId>164341c-9708-4967-b3ec-24933767fd44</AppId>
      <CoTypeCode>CoCreateApp</CoTypeCode>
      <FinishTime>2019-07-11 20:12:58</FinishTime>
      <CoType>创建应用</CoType>
      <BatchCount>1</BatchCount>
    </ChangeOrderList>
    <TotalSize>1</TotalSize>
  </Data>
  <ErrorCode>success</ErrorCode>
  <Code>200</Code>
  <Success>true</Success>
</ListChangeOrders>
```

### JSON 格式

```
{
  "Data": {
    "ChangeOrderList": [
      {
        "Source": "console",
        "Status": 2,
        "Description": "版本: 1.0 | 镜像名称: nginx ",
        "BatchCount": 1,
        "CoTypeCode": "CoCreateApp",
        "FinishTime": "2019-07-11 20:12:58",
        "BatchType": "自动",
        "CreateTime": "2019-07-11 15:54:49",
        "CoType": "创建应用",
        "AppId": "164341c-9708-4967-b3ec-24933767fd44",
        "ChangeOrderId": "7fa5c0-9ebb-4bb4-b383-1f885447b109",

```

```
"GroupId": "c9ecd2-cf6c-46c3-9f20-525de202cf79"
  }
],
"PageSize": 20,
"TotalSize": 1,
"CurrentPage": 1
},
"Message": "success",
"RequestId": "65E1F-43BA-4D0C-8E61-E4D1337FA9B2",
"TraceId": "0bb6f815638568884597879d60be",
"Success": true,
"ErrorCode": "success",
"Code": 200
}
```

### 错误码

HttpCode	错误码	错误信息	描述
400	InvalidParameter. NotEmpty	You must specify the parameter %s.	不合法的参数: %s 不能为空
400	Resouce.no. permission	You are not authorized to operate on the specified resources.	没有权限操作资源

访问[错误中心](#)查看更多错误码。

## 8.10 AbortAndRollbackChangeOrder

中止或回滚变更单

##请求语法

### 调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

### 请求头

该接口使用公共请求头，无特殊请求头。请参见公共请求参数文档。



## 请求语法

```
PUT /pop/v5/changeorder/ChangeOrderAbortAndRollback HTTP/1.1
```

## 请求参数

名称	类型	是否必选	示例值	描述
ChangeOrderId	String	是	xxx-xxxx-xxx-xxxx	变更单ID

## 返回数据

名称	类型	示例值	描述
Code	String	200	接口状态或 POP 错误码
Data			变更单信息
ChangeOrderId	String	xxx-xxxx-xxx-xxxx	变更单ID
ErrorCode	String	success	错误码
Message	String	success	附加信息
RequestId	String	00000000-0000-0000-0000-000000000000	请求ID
Success	Boolean	true	是否成功
TraceId	String	0000000000 0000000000 0000000000	调用链ID

## 示例

## 请求示例

```
PUT /pop/v5/changeorder/ChangeOrderAbortAndRollback HTTP/1.1
公共请求头
{
  "ChangeOrderId": "xxx-xxxx-xxx-xxxx"
}
```

## 正常返回示例

## XML 格式

```
<AbortAndRollbackChangeOrder>
  <Data>
    <ChangeOrderId>xxx-xxxx-xxx-xxxx</ChangeOrderId>
  </Data>
  <Message>success</Message>
  <TraceId>00000000000000000000000000000000</TraceId>
  <RequestId>00000000-0000-0000-0000-000000000000</RequestId>
  <Success>>true</Success>
  <ErrorCode>success</ErrorCode>
  <Code>200</Code>
</AbortAndRollbackChangeOrder>
```

## JSON 格式

```
{
  "Data":{
    "ChangeOrderId":"xxx-xxxx-xxx-xxxx"
  },
  "Message":"success",
  "RequestId":"00000000-0000-0000-0000-000000000000",
  "TraceId":"00000000000000000000000000000000",
  "ErrorCode":"success",
  "Success":true,
  "Code":200
}
```

## 错误码

HttpCode	错误码	错误信息	描述
400	InvalidParameter. NotEmpty	You must specify the parameter %s.	不合法的参数: %s 不能为空
400	Resouce.no.permission	You are not authorized to operate on the specified resources.	没有权限操作资源
400	Application.ChangerOrderRunning	An application change process is in progress. Please try again later.	应用有变更流程正在执行, 请稍后重试。
400	InvalidChangeOrder. NotFound	The current change order does not exist.	变更单不存在

访问[错误中心](#)查看更多错误码。

## 8.11 DescribeChangeOrder

查询变更单信息

调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

请求头

该接口使用公共请求头，无特殊请求头。请参见公共请求参数文档。

请求语法

```
GET /pop/v1/sam/changeorder/DescribeChangeOrder HTTP/1.1
```

请求参数

名称	类型	是否必选	示例值	描述
ChangeOrderId	String	是	76fa5c0-9ebb-4bb4-b383-1f885447b109	变更单ID

返回数据

名称	类型	示例值	描述
Code	String	200	接口状态或 POP 错误码
Data			变更单信息
AppName	String	app-test	应用名称
Auto	Boolean	true	是否为自动分批
BatchCount	Integer	1	分批数
BatchType	String	自动	分批类型
BatchWaitTime	Integer	0	自动分批方式，等待多长时间开始下一批次，单位：分钟

名称	类型	示例值	描述
ChangeOrderId	String	765fa5c0-9ebb-4bb4-b383-1f885447b109	变更单ID
CoType	String	变更类型	创建应用
CoTypeCode	String	CoCreateApp	变更类型CODE
CreateTime	String	2019-07-11 15:54:49	创建时间
CurrentPipelineId	String	06-ba2d-46ea-9e79-0727fbca6eda	当前批次ID
Description	String	版本: 1.0   镜像名称: nginx	变更单描述信息
ErrorMessage	String	success	错误信息
Pipelines			批次信息
BatchType	Integer	0	分批类型
ParallelCount	Integer	0	分批内并行任务数
PipelineId	String	0cffd-ba2d-46ea-9e79-0727fbca	批次ID
PipelineName	String	第1批变更	批次名称
StartTime	Long	1562831689704	开始时间
Status	Integer	2	批次状态, 0: 准备 1: 执行中 2: 执行成功 3: 执行失败 6: 终止 10: 系统异常执行失败
UpdateTime	Long	1562847178007	最近更新时间
Status	Integer	2	变更单状态, 0: 准备 1: 执行中 2: 执行成功 3: 执行失败 6: 终止 10: 系统异常执行失败

名称	类型	示例值	描述
SupportRollback	Boolean	false	是否支持回滚
ErrorCode	String	success	错误码
Message	String	success	附加信息
RequestId	String	00000000-0000-0000-0000-000000000000	请求ID
Success	Boolean	true	是否成功
TraceId	String	0000000000 0000000000 0000000000	调用链ID

## 示例

### 请求示例

```
GET /pop/v1/sam/changeorder/DescribeChangeOrder HTTP/1.1
公共请求头
{
  "ChangeOrderId": "76fa5c0-9ebb-4bb4-b383-1f885447b109"
}
```

### 正常返回示例

#### XML 格式

```
<DescribeChangeOrder>
  <Message>success</Message>
  <RequestId>XXXC03A-E74E-467E-B4E2-63E3F5AACF55</RequestId>
  <TraceId>0bc3915638507554994370d5056</TraceId>
  <Data>
    <Status>2</Status>
    <Description>版本: 1.0 | 镜像名称: nginx </Description>
    <Pipelines>
      <Status>2</Status>
      <PipelineName>第1批变更</PipelineName>
      <ParallelCount>0</ParallelCount>
      <UpdateTime>1562847178007</UpdateTime>
      <StartTime>1562831689704</StartTime>
      <PipelineId>0cffd-ba2d-46ea-9e79-0727fbca6eda</PipelineId>
      <BatchType>0</BatchType>
    </Pipelines>
    <CreateTime>2019-07-11 15:54:49</CreateTime>
    <ChangeOrderId>765fa5c0-9ebb-4bb4-b383-1f885447b109</ChangeOrder
  rId>
    <BatchType>自动</BatchType>
    <AppName>app-test</AppName>
```

```

    <Auto>true</Auto>
    <CurrentPipelineId>06-ba2d-46ea-9e79-0727fbca6eda</CurrentPipelineId>
    <CoTypeCode>CoCreateApp</CoTypeCode>
    <SupportRollback>>false</SupportRollback>
    <BatchWaitTime>0</BatchWaitTime>
    <CoType>创建应用</CoType>
    <BatchCount>1</BatchCount>
  </Data>
  <ErrorCode>success</ErrorCode>
  <Code>200</Code>
  <Success>>true</Success>
</DescribeChangeOrder>

```

## JSON 格式

```

{
  "Data": {
    "Description": "版本: 1.0 | 镜像名称: nginx ",
    "BatchType": "自动",
    "CoType": "创建应用",
    "ChangeOrderId": "765fa5c0-9ebb-4bb4-b383-1f885447b109",
    "SupportRollback": false,
    "Status": 2,
    "BatchCount": 1,
    "CoTypeCode": "CoCreateApp",
    "CreateTime": "2019-07-11 15:54:49",
    "Auto": true,
    "AppName": "app-test",
    "Pipelines": [
      {
        "ParallelCount": 0,
        "Status": 2,
        "PipelineName": "第1批变更",
        "BatchType": 0,
        "PipelineId": "0cffd-ba2d-46ea-9e79-0727fbca6eda",
        "UpdateTime": 1562847178007,
        "StartTime": 1562831689704
      }
    ],
    "CurrentPipelineId": "06-ba2d-46ea-9e79-0727fbca6eda",
    "BatchWaitTime": 0
  },
  "Message": "success",
  "RequestId": "XXXC03A-E74E-467E-B4E2-63E3F5AACF55",
  "TraceId": "0bc3915638507554994370d5056",
  "Success": true,
  "ErrorCode": "success",
  "Code": 200
}

```

## 错误码

HttpCode	错误码	错误信息	描述
400	InvalidParameter. NotEmpty	You must specify the parameter %s.	不合法的参数: %s 不能为空

HttpCode	错误码	错误信息	描述
400	Resouce.no.permission	You are not authorized to operate on the specified resources.	没有权限操作资源
400	InvalidChangeOrder.NotFound	The current change order does not exist.	变更单不存在

访问[错误中心](#)查看更多错误码。

## 8.12 DescribeInstanceSpecifications

获取应用实例规格信息

调试

您可以在[OpenAPI Explorer](#)中直接运行该接口，免去您计算签名的困扰。运行成功后，[OpenAPI Explorer](#)可以自动生成SDK代码示例。

请求头

该接口使用公共请求头，无特殊请求头。请参见公共请求参数文档。

请求语法

```
GET /pop/v1/paas/quota/instanceSpecifications HTTP/1.1
```

请求参数

返回数据

名称	类型	示例值	描述
Code	String	200	接口状态或 POP 错误码
Data			实例规格信息
Cpu	Integer	2000	cpu大小，规格：微核
Enable	Boolean	true	是否可用
Id	Integer	4	规格配置ID
Memory	Integer	4096	内存配置，规格：MB

名称	类型	示例值	描述
SpecInfo	String	通用型4	规格配置名称
Version	Integer	0	规格配置版本号
ErrorCode	String	success	错误码
Message	String	success	附加信息
RequestId	String	00000000-0000-0000-0000-000000000000	请求ID
Success	Boolean	true	是否成功
TraceId	String	0000000000 0000000000 0000000000	调用链ID

## 示例

### 请求示例

```
GET /pop/v1/paas/quota/instanceSpecifications HTTP/1.1
公共请求头
```

### 正常返回示例

#### XML 格式

```
<DescribeInstanceSpecifications>
  <Message>success</Message>
  <RequestId>3666F8BD-475B-4A0F-813D-330660965</RequestId>
  <TraceId>0bc3b6e415638596990245792de7</TraceId>
  <Data>
    <SpecInfo>通用型1</SpecInfo>
    <Version>0</Version>
    <Memory>1024</Memory>
    <Cpu>1000</Cpu>
    <Enable>>false</Enable>
    <Id>1</Id>
  </Data>
  <Data>
    <SpecInfo>通用型2</SpecInfo>
    <Version>0</Version>
    <Memory>2048</Memory>
    <Cpu>1000</Cpu>
    <Enable>>true</Enable>
    <Id>2</Id>
  </Data>
  <ErrorCode>success</ErrorCode>
  <Code>200</Code>
```



```
<Success>true</Success>
</DescribeInstanceSpecifications>
```

### JSON 格式

```
{
  "Data": [
    {
      "Cpu": 1000,
      "Memory": 1024,
      "Enable": false,
      "SpecInfo": "通用型1",
      "Id": 1,
      "Version": 0
    },
    {
      "Cpu": 1000,
      "Memory": 2048,
      "Enable": true,
      "SpecInfo": "通用型2",
      "Id": 2,
      "Version": 0
    }
  ],
  "Message": "success",
  "RequestId": "3666F8BD-475B-4A0F-813D-330660965",
  "TraceId": "0bc3b6e415638596990245792de7",
  "Success": true,
  "ErrorCode": "success",
  "Code": 200
}
```

### 错误码

访问[错误中心](#)查看更多错误码。

## 8.13 RescaleApplication

### 应用扩缩

### 调试

您可以在[OpenAPI Explorer](#)中直接运行该接口，免去您计算签名的困扰。运行成功后，[OpenAPI Explorer](#)可以自动生成SDK代码示例。

### 请求头

该接口使用公共请求头，无特殊请求头。请参见公共请求参数文档。

## 请求语法

```
PUT /pop/v1/sam/app/rescaleApplication HTTP/1.1
```

## 请求参数

名称	类型	是否必选	示例值	描述
AppId	String	是	0099b7be-5f5b-4512-a7fc-56049ef1aa6b	目标应用ID
Replicas	Integer	是	2	目标实例数

## 返回数据

名称	类型	示例值	描述
Code	String	200	接口状态或 POP 错误码
Data			返回结果
ChangeOrderId	String	0e09829f-4914-4612-bc88-6f52fd813c33	发布单ID
ErrorCode	String	success	错误码
Message	String	success	附加信息
RequestId	String	00000000-0000-0000-000000000000	请求ID
Success	Boolean	true	是否成功

## 示例

## 请求示例

```
PUT /pop/v1/sam/app/rescaleApplication HTTP/1.1
公共请求头
{
  "AppId": "0099b7be-5f5b-4512-a7fc-56049ef1aa6b"
  "Replicas": "2"
}
```

## 正常返回示例

## XML 格式

```
<RescaleApplicationResponse>
  <Message>success</Message>
  <RequestId>00000000-0000-0000-0000-000000000000</RequestId>
  <Data>
    <ChangeOrderId>0e09829f-4914-4612-bc88-6f52fd813c33</ChangeOrder
    rId>
  </Data>
  <ErrorCode>success</ErrorCode>
  <Code>200</Code>
  <Success>>true</Success>
</RescaleApplicationResponse>
```

## JSON 格式

```
{
  "Data":{
    "ChangeOrderId":"0e09829f-4914-4612-bc88-6f52fd813c33"
  },
  "Message":"success",
  "RequestId":"00000000-0000-0000-0000-000000000000",
  "Success":true,
  "ErrorCode":"success",
  "Code":200
}
```

## 错误码

HttpCode	错误码	错误信息	描述
400	Application.ChangerOrderRunning	An application change process is in progress. Please try again later.	应用有变更流程正在执行, 请稍后重试。
400	Application.InvalidStatus	The application status is abnormal. Please try again later.	应用状态异常, 请稍后重试。
400	Application.NotDeployYet	The application has not been deployed. Please deploy it and try again.	应用没有部署, 请部署后重试。
400	InvalidApplication.NotFound	The current application does not exist.	找不到当前应用
400	InvalidParameter.NotEmpty	You must specify the parameter %s.	不合法的参数: %s 不能为空
400	InvalidParameter.Obviously	The specified parameter is invalid {%s}.	不合法的参数{%s}

HttpCode	错误码	错误信息	描述
400	InvalidParameter. WithMessage	The parameter is invalid {%s}: %s	不合法的参数{%s}: %s

访问[错误中心](#)查看更多错误码。

## 8.14 RestartApplication

重启应用

调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

请求头

该接口使用公共请求头，无特殊请求头。请参见公共请求参数文档。

请求语法

```
PUT /pop/v1/sam/app/restartApplication HTTP/1.1
```

请求参数

名称	类型	是否必选	示例值	描述
AppId	String	是	0099b7be-5f5b-4512-a7fc-56049ef1aa6b	目标应用ID
MinReadyInstances	Integer	否	1	最小可用实例数。在变更期间保证可用的实例数。

返回数据

名称	类型	示例值	描述
Code	String	200	接口状态或 POP 错误码
Data			返回结果
ErrorCode	String	success	错误码

名称	类型	示例值	描述
Message	String	success	附加信息
RequestId	String	00000000-0000 -0000-0000- 000000000000	请求ID
Success	Boolean	true	是否成功
TraceId	String	0000000000 0000000000 0000000000	调用链ID

## 示例

### 请求示例

```
PUT /pop/v1/sam/app/restartApplication HTTP/1.1
公共请求头
{
  "AppId": "0099b7be-5f5b-4512-a7fc-56049ef1aa6b"
}
```

### 正常返回示例

#### XML 格式

```
<RestartApplicationResponse>
  <Message>success</Message>
  <RequestId>7BD8F4C7-D84C-4D46-9885-8212997E24D6</RequestId>
  <TraceId>0bc3b6e215637275918588187d221e</TraceId>
  <Data>
    <ChangeOrderId>4a815998-b468-4bea-b7d8-59f52a4421f5</ChangeOrder
    rId>
  </Data>
  <ErrorCode>success</ErrorCode>
  <Code>200</Code>
  <Success>>true</Success>
</RestartApplicationResponse>
```

#### JSON 格式

```
{
  "Data": {
    "ChangeOrderId": "4a815998-b468-4bea-b7d8-59f52a4421f5"
  },
  "Message": "success",
  "RequestId": "7BD8F4C7-D84C-4D46-9885-8212997E24D6",
  "TraceId": "0bc3b6e215637275918588187d221e",
  "Success": true,
  "ErrorCode": "success",
  "Code": 200
}
```

```
}

```

## 错误码

HttpCode	错误码	错误信息	描述
400	Application.ChangerOrderRunning	An application change process is in progress. Please try again later.	应用有变更流程正在执行, 请稍后重试。
400	Application.InvalidStatus	The application status is abnormal. Please try again later.	应用状态异常, 请稍后重试。
400	Application.NotDeployYet	The application has not been deployed. Please deploy it and try again.	应用没有部署, 请部署后重试。
400	InvalidApplication.NotFound	The current application does not exist.	找不到当前应用
400	InvalidParameter.NotEmpty	You must specify the parameter %s.	不合法的参数: %s 不能为空
400	InvalidParameter.Obviously	The specified parameter is invalid {%s}.	不合法的参数{%s}
400	InvalidParameter.WithMessage	The parameter is invalid {%s}: %s	不合法的参数{%s}: %s
400	NoComputeResourceQuota.Exceed	Your compute resource is insufficient. Please submit a ticket to raise the quota.	计算资源不足, 请提交工单增加计算资源额度。

访问[错误中心](#)查看更多错误码。

## 8.15 AbortChangeOrder

中止变更单

### 调试

您可以在OpenAPI Explorer中直接运行该接口, 免去您计算签名的困扰。运行成功后, OpenAPI Explorer可以自动生成SDK代码示例。

## 请求头

该接口使用公共请求头，无特殊请求头。请参见公共请求参数文档。

## 请求语法

```
PUT /pop/v1/sam/changeorder/AbortChangeOrder HTTP/1.1
```

## 请求参数

名称	类型	是否必选	示例值	描述
ChangeOrderId	String	是	xxx-xxxx-xxx-xxx	变更单ID

## 返回数据

名称	类型	示例值	描述
Code	String	200	接口状态或 POP 错误码
Data			变更单信息
ChangeOrderId	String	xxx-xxxx-xxx-xxx	变更单ID
ErrorCode	String	success	错误码
Message	String	success	附加信息
RequestId	String	00000000-0000-0000-0000-000000000000	请求ID
Success	Boolean	true	是否成功
TraceId	String	0000000000 0000000000 0000000000	调用链ID

## 示例

## 请求示例

```
PUT /pop/v1/sam/changeorder/AbortChangeOrder HTTP/1.1
公共请求头
{
  "ChangeOrderId": "xxx-xxxx-xxx-xxx"
```

```
}

```

## 正常返回示例

### XML 格式

```
<AbortChangeOrder>
  <Data>
    <ChangeOrderId>xxx-xxxx-xxx-xxx</ChangeOrderId>
  </Data>
  <Message>success</Message>
  <TraceId>00000000000000000000000000000000</TraceId>
  <RequestId>00000000-0000-0000-0000-000000000000</RequestId>
  <Success>>true</Success>
  <ErrorCode>success</ErrorCode>
  <Code>200</Code>
</AbortChangeOrder>

```

### JSON 格式

```
{
  "Data": {
    "ChangeOrderId": "xxx-xxxx-xxx-xxx"
  },
  "Message": "success",
  "RequestId": "00000000-0000-0000-0000-000000000000",
  "TraceId": "00000000000000000000000000000000",
  "ErrorCode": "success",
  "Success": true,
  "Code": 200
}
```

## 错误码

HttpCode	错误码	错误信息	描述
400	InvalidParameter. NotEmpty	You must specify the parameter %s.	不合法的参数: %s 不能为空
400	Resouce.no. permission	You are not authorized to operate on the specified resources.	没有权限操作资源
400	Application. ChangerOrderRunning	An application change process is in progress. Please try again later.	应用有变更流程正在执行, 请稍后重试。

访问[错误中心](#)查看更多错误码。



## 9 SLB

### 9.1 BindSlb

为应用绑定SLB

调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

请求头

该接口使用公共请求头，无特殊请求头。请参见公共请求参数文档。

请求语法

```
POST /pop/v1/sam/app/slb HTTP/1.1
```

请求参数

名称	类型	是否必选	示例值	描述
AppId	String	是	0099b7be-5f5b-4512-a7fc-56049ef1aa6b	部署成功的目标应用ID
Internet	String	否	{ "port":80,"targetPort":8080,"protocol":"TCP" }	绑定公网SLB。例如: { "port":80,"targetPort":8080,"protocol":"TCP" }，表示将容器的8080端口通过slb的80端口暴露服务，协议为TCP，为空忽略。
Intranet	String	否	{ "port":80,"targetPort":8080,"protocol":"TCP" }	绑定私网SLB。例如: { "port":80,"targetPort":8080,"protocol":"TCP" }，表示将容器的8080端口通过slb的80端口暴露服务，协议为TCP，为空忽略。

## 返回数据

名称	类型	示例值	描述
Code	String	200	接口状态或 POP 错误码
Data			返回结果
ChangeOrderId	String	01db03d3-3ee9-48b3-b3d0-dfce2d88ab31	返回的发布单ID，用于查询任务执行状态。
ErrorCode	String	success	错误码
Message	String	success	附加信息
RequestId	String	00000000-0000-0000-0000-000000000000	请求ID
Success	Boolean	true	是否成功
TraceId	String	0000000000 0000000000 0000000000	调用链ID

## 示例

## 请求示例

```
POST /pop/v1/sam/app/slb HTTP/1.1
公共请求头
{
  "AppId": "0099b7be-5f5b-4512-a7fc-56049ef1aa6b"
}
```

## 正常返回示例

## XML 格式

```
<BindSlb>
  <Data>
    <ChangeOrderId>01db03d3-3ee9-48b3-b3d0-dfce2d88ab31</ChangeOrderId>
  </Data>
  <Message>success</Message>
  <RequestId>00000000-0000-0000-0000-000000000000</RequestId>
  <TraceId>00000000000000000000000000000000</TraceId>
  <Success>true</Success>
  <ErrorCode>success</ErrorCode>
  <Code>200</Code>
```

&lt;/BindSlb&gt;

## JSON 格式

```
{
  "Data":{
    "ChangeOrderId":"01db03d3-3ee9-48b3-b3d0-dfce2d88ab31"
  },
  "Message":"success",
  "RequestId":"00000000-0000-0000-0000-000000000000",
  "TraceId":"00000000000000000000000000000000",
  "ErrorCode":"success",
  "Success":true,
  "Code":200
}
```

## 错误码

HttpCode	错误码	错误信息	描述
400	Application.ChangerOrderRunning	An application change process is in progress. Please try again later.	应用有变更流程正在执行，请稍后重试。
400	Application.InvalidStatus	The application status is abnormal. Please try again later.	应用状态异常，请稍后重试。
400	Application.NotDeployYet	The application has not been deployed. Please deploy it and try again.	应用没有部署，请部署后重试。
400	InvalidApplication.NotFound	The current application does not exist.	找不到当前应用
400	InvalidParameter.ProtocolNotSupport	Only TCP protocol is supported.	不支持指定的协议，只支持TCP协议。
400	InvalidParameter.NotEmpty	You must specify the parameter %s.	不合法的参数：%s 不能为空
400	InvalidParameter.Obviously	The specified parameter is invalid %s}.	不合法的参数{%s}

访问[错误中心](#)查看更多错误码。

## 9.2 UnbindSlb

解绑内网或公网SLB

调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

请求头

该接口使用公共请求头，无特殊请求头。请参见公共请求参数文档。

请求语法

```
DELETE /pop/v1/sam/app/slb HTTP/1.1
```

请求参数

名称	类型	是否必选	示例值	描述
AppId	String	是	0099b7be-5f5b-4512-a7fc-56049ef1aa6b	目标应用ID
Internet	Boolean	是	true	是否解绑公网SLB
Intranet	Boolean	是	true	是否解绑私网SLB

返回数据

名称	类型	示例值	描述
Code	String	200	接口状态或 POP 错误码
Data			返回结果
ChangeOrderId	String	4a815998-b468-4bea-b7d8-59f52a4421f5	发布单ID
ErrorCode	String	success	错误码
Message	String	success	附加信息

名称	类型	示例值	描述
RequestId	String	00000000-0000-0000-0000-000000000000	请求ID
Success	Boolean	true	是否成功
TraceId	String	0000000000 0000000000 0000000000	调用链ID

## 示例

### 请求示例

```
DELETE /pop/v1/sam/app/slb HTTP/1.1
公共请求头
{
  "AppId": "0099b7be-5f5b-4512-a7fc-56049ef1aa6b"
  "Internet": "true"
  "Intranet": "true"
}
```

### 正常返回示例

#### XML 格式

```
<UnbindSlbResponse>
  <Message>success</Message>
  <RequestId>7BD8F4C7-D84C-4D46-9885-8212997E24D6</RequestId>
  <TraceId>0bc3b6e215637275918588187d221e</TraceId>
  <Data>
    <ChangeOrderId>4a815998-b468-4bea-b7d8-59f52a4421f5</ChangeOrder
rId>
  </Data>
  <ErrorCode>success</ErrorCode>
  <Code>200</Code>
  <Success>>true</Success>
</UnbindSlbResponse>
```

#### JSON 格式

```
{
  "Data":{
    "ChangeOrderId": "4a815998-b468-4bea-b7d8-59f52a4421f5"
  },
  "Message": "success",
  "RequestId": "7BD8F4C7-D84C-4D46-9885-8212997E24D6",
  "TraceId": "0bc3b6e215637275918588187d221e",
  "Success": true,
  "ErrorCode": "success",
  "Code": 200
}
```

}

## 错误码

HttpCode	错误码	错误信息	描述
400	Application.ChangerOrderRunning	An application change process is in progress. Please try again later.	应用有变更流程正在执行, 请稍后重试。
400	Application.InvalidStatus	The application status is abnormal. Please try again later.	应用状态异常, 请稍后重试。
400	Application.NotDeployYet	The application has not been deployed. Please deploy it and try again.	应用没有部署, 请部署后重试。
400	InvalidApplication.NotFound	The current application does not exist.	找不到当前应用
400	InvalidParameter.NotEmpty	You must specify the parameter %s.	不合法的参数: %s 不能为空
400	InvalidParameter.Obviously	The specified parameter is invalid %s}.	不合法的参数{%s}
400	InvalidParameter.WithMessage	The parameter is invalid %s}: %s	不合法的参数{%s}: %s

访问[错误中心](#)查看更多错误码。

## 9.3 DescribeApplicationSlbs

获取应用SLB配置信息

### 调试

您可以在OpenAPI Explorer中直接运行该接口, 免去您计算签名的困扰。运行成功后, OpenAPI Explorer可以自动生成SDK代码示例。

### 请求头

该接口使用公共请求头, 无特殊请求头。请参见公共请求参数文档。

## 请求语法

```
GET /pop/v1/sam/app/slb HTTP/1.1
```

## 请求参数

名称	类型	是否必选	示例值	描述
AppId	String	是	017f39b8-dfa4-4e16-a84b-1dcee4b17106	目标应用ID

## 返回数据

名称	类型	示例值	描述
Code	String	200	接口状态或 POP 错误码
Data			返回结果
Internet			公网SLB配置
Port	Integer	80	SLB监听端口
Protocol	String	TCP	支持的协议
TargetPort	Integer	8080	容器端口
InternetIp	String	59.74.23.155	公网SLB地址
Intranet			内网SLB配置
Port	Integer	80	SLB监听端口
Protocol	String	TCP	支持的协议
TargetPort	Integer	8080	容器端口
IntranetIp	String	192.168.2.66	内网SLB地址

## 示例

## 请求示例

```
GET /pop/v1/sam/app/slb HTTP/1.1
```

**公共请求头**

```
{
  "AppId": "017f39b8-dfa4-4e16-a84b-1dcee4b17106"
}
```

**正常返回示例****XML 格式**

```
<DescribeApplicationSlbsResponse>
  <Message>success</Message>
  <RequestId>7CFB4988-7267-4397-A648-0941471AA0D5</RequestId>
  <TraceId>0bc3b6e215639348698488298d221e</TraceId>
  <Data>
    <InternetIp>112.124.199.136</InternetIp>
    <Intranet>
      <TargetPort>8080</TargetPort>
      <Port>80</Port>
      <Protocol>TCP</Protocol>
    </Intranet>
    <Internet>
      <TargetPort>8080</TargetPort>
      <Port>65530</Port>
      <Protocol>TCP</Protocol>
    </Internet>
    <IntranetIp>192.168.0.29</IntranetIp>
  </Data>
  <ErrorCode>success</ErrorCode>
  <Code>200</Code>
  <Success>>true</Success>
</DescribeApplicationSlbsResponse>
```

**JSON 格式**

```
{
  "Data": {
    "IntranetIp": "192.168.0.29",
    "InternetIp": "112.124.199.136",
    "Intranet": [
      {
        "Port": 80,
        "TargetPort": 8080,
        "Protocol": "TCP"
      }
    ],
    "Internet": [
      {
        "Port": 65530,
        "TargetPort": 8080,
        "Protocol": "TCP"
      }
    ]
  },
  "Message": "success",
  "RequestId": "7CFB4988-7267-4397-A648-0941471AA0D5",
  "TraceId": "0bc3b6e215639348698488298d221e",
  "Success": true,
  "ErrorCode": "success",
  "Code": 200
}
```



```
}
```

## 错误码

HttpCode	错误码	错误信息	描述
400	InvalidApplication.NotFound	The current application does not exist.	找不到当前应用
400	InvalidParameter.NotEmpty	You must specify the parameter %s.	不合法的参数: %s 不能为空
400	InvalidParameter.Obviously	The specified parameter is invalid {%s}.	不合法的参数{%s}
400	InvalidParameter.WithMessage	The parameter is invalid {%s}: %s	不合法的参数{%s}: %s

访问[错误中心](#)查看更多错误码。

## 10 微服务列表

### 10.1 ListConsumedServices

获取订阅的微服务列表

调试

您可以在OpenAPI Explorer中直接运行该接口，免去您计算签名的困扰。运行成功后，OpenAPI Explorer可以自动生成SDK代码示例。

请求头

该接口使用公共请求头，无特殊请求头。请参见公共请求参数文档。

请求语法

```
GET /pop/v1/sam/service/listConsumedServices HTTP/1.1
```

请求参数

名称	类型	是否必选	示例值	描述
AppId	String	是	00000000-0000-0000-0000-000000000000	应用ID

返回数据

名称	类型	示例值	描述
Code	String	200	接口状态或 POP 错误码
Data			微服务列表信息
AppId	String	00000000-0000-0000-0000-000000000000	应用ID
Group2Ip	String	{}	保留字段
Groups		HSF	消费的服务对应的组别
Ips		10.0.0.1	服务订阅地址

名称	类型	示例值	描述
Name	String	com.alibaba .nodejs. ItemService	发布的服务名称
Type	String	RPC	发布的服务类型
Version	String	1.0.0	发布的服务版本
ErrorCode	String	success	错误码
Message	String	success	附加信息
RequestId	String	00000000-0000 -0000-0000- 000000000000	请求ID
Success	Boolean	true	是否成功
TraceId	String	0000000000 0000000000 0000000000	调用链ID

## 示例

### 请求示例

```
GET /pop/v1/sam/service/listConsumedServices HTTP/1.1
公共请求头
{
  "AppId": "00000000-0000-0000-0000-000000000000"
}
```

### 正常返回示例

#### XML 格式

```
<ListConsumedServicesResponse>
  <Data>
    <Groups>HSF</Groups>
    <Name>com.alibaba.nodejs.ItemService</Name>
    <Type>RPC</Type>
    <Group2Ip></Group2Ip>
    <Version>1.0.0</Version>
  </Data>
  <Message>success</Message>
  <RequestId>00000000-0000-0000-0000-000000000000</RequestId>
  <TraceId>00000000000000000000000000000000</TraceId>
  <ErrorCode>success</ErrorCode>
  <Success>>true</Success>
  <Code>200</Code>
```

```
</ListConsumedServicesResponse>
```

### JSON 格式

```
{
  "Data": [
    {
      "Name": "com.alibaba.nodejs.ItemService",
      "Groups": [
        "HSF"
      ],
      "Type": "RPC",
      "Group2Ip": {},
      "Version": "1.0.0",
      "Ips": []
    }
  ],
  "Message": "success",
  "RequestId": "00000000-0000-0000-0000-000000000000",
  "TraceId": "00000000000000000000000000000000",
  "Success": true,
  "ErrorCode": "success",
  "Code": 200
}
```

### 错误码

访问[错误中心](#)查看更多错误码。

## 10.2 ListPublishedServices

获取发布的微服务列表

### 调试

您可以在[OpenAPI Explorer](#)中直接运行该接口，免去您计算签名的困扰。运行成功后，[OpenAPI Explorer](#)可以自动生成SDK代码示例。

### 请求头

该接口使用公共请求头，无特殊请求头。请参见公共请求参数文档。

### 请求语法

```
GET /pop/v1/sam/service/listPublishedServices HTTP/1.1
```

### 请求参数

名称	类型	是否必选	示例值	描述
AppId	String	是	00000000-0000-0000-0000-000000000000	应用ID

## 返回数据

名称	类型	示例值	描述
Code	String	200	接口状态或 POP 错误码
Data			微服务列表信息
AppId	String	00000000-0000-0000-0000-000000000000	应用ID
Group2Ip	String	{}	保留字段
Groups		HSF	消费的服务对应的组别
Ips		10.0.0.1	服务订阅地址
Name	String	com.alibaba.nodejs.ItemService	发布的服务名称
Type	String	RPC	发布的服务类型
Version	String	1.0.0	发布的服务版本
ErrorCode	String	success	错误码
Message	String	success	附加信息
RequestId	String	00000000-0000-0000-0000-000000000000	请求ID
Success	Boolean	true	是否成功
TraceId	String	0000000000 0000000000 0000000000	调用链ID

## 示例

## 请求示例

```
GET /pop/v1/sam/service/listPublishedServices HTTP/1.1
公共请求头
{
```

```
"AppId": "00000000-0000-0000-0000-000000000000"
}
```

## 正常返回示例

### XML 格式

```
<ListPublishedServicesResponse>
  <Data>
    <Groups>HSF</Groups>
    <Name>com.alibaba.nodejs.ItemService</Name>
    <Type>RPC</Type>
    <Group2Ip></Group2Ip>
    <Version>1.0.0</Version>
  </Data>
  <Message>success</Message>
  <RequestId>00000000-0000-0000-0000-000000000000</RequestId>
  <TraceId>00000000000000000000000000000000</TraceId>
  <ErrorCode>success</ErrorCode>
  <Success>true</Success>
  <Code>200</Code>
</ListPublishedServicesResponse>
```

### JSON 格式

```
{
  "Data": [
    {
      "Name": "com.alibaba.nodejs.ItemService",
      "Groups": [
        "HSF"
      ],
      "Type": "RPC",
      "Group2Ip": {},
      "Version": "1.0.0",
      "Ips": []
    }
  ],
  "Message": "success",
  "RequestId": "00000000-0000-0000-0000-000000000000",
  "TraceId": "00000000000000000000000000000000",
  "Success": true,
  "ErrorCode": "success",
  "Code": 200
}
```

## 错误码

访问[错误中心](#)查看更多错误码。

## 11 公共错误码

HttpCode	错误码	错误信息	描述
404	InvalidResponse.Api	The response of API %s is empty.	API 返回数据为空
400	Application.InvalidStatus	The application status is abnormal. Please try again later.	应用状态异常，请稍后重试。
404	InvalidAppId.NotFound	The specified AppId does not exist.	指定的AppId不存在。
400	InvalidApplication.NotFound	The current application does not exist.	找不到当前应用
400	InvalidChangeOrder.NotFound	The current change order does not exist.	变更单不存在
400	InvalidComponent.NotFound	The current component (such as JDK, Tomcat, or EDASWebContainer) does not exist.	找不到当前组件 (JDK、Tomcat、EDASWebContainer等)
404	InvalidNamespaceId.NotFound	The specified NamespaceId does not exist.	指定的NamespaceId不存在。
400	InvalidPackageType.NotFound	The package type must be War, FatJar, or Image.	包类型必须为War、FatJar或Image
400	InvalidParameter.ProtocolNotSupport	Only TCP protocol is supported.	不支持指定的协议，只支持TCP协议。
400	InvalidParameter.NotEmpty	You must specify the parameter %s.	不合法的参数：%s 不能为空
400	InvalidParameter.Obviously	The specified parameter is invalid {%s}.	不合法的参数{%s}

HttpCode	错误码	错误信息	描述
400	InvalidParameter. WithMessage	The parameter is invalid {%s}: %s	不合法的参数{%s}: %s
400	InvalidServerlessRegion. Unsupported	The current region is not supported: %s	不支持当前地域: %s
404	InvalidVpcId. NotFound	The specified VpcId does not exist.	指定的VpcId不存在。
400	NoAvailableCluster. NotFound	No clusters are available for the current region.	当前地域没有可用集群

访问[错误中心](#)查看更多错误码。