

阿里云 访问控制

API参考 (RAM)

文档版本：20190919

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。	 禁止： 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。	 警告： 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。	 说明： 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
粗体	表示按键、菜单、页面名称等UI元素。	单击 确定 。
<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 调用方式.....	1
1.1 请求结构.....	1
1.2 公共参数.....	1
1.3 计算签名.....	3
1.4 返回结果.....	5
2 RAM鉴权.....	7
3 API概览.....	11
4 用户管理接口.....	14
4.1 CreateUser.....	14
4.2 GetUser.....	16
4.3 UpdateUser.....	18
4.4 DeleteUser.....	20
4.5 ListUsers.....	22
4.6 CreateLoginProfile.....	24
4.7 GetLoginProfile.....	26
4.8 DeleteLoginProfile.....	28
4.9 UpdateLoginProfile.....	29
4.10 CreateAccessKey.....	30
4.11 UpdateAccessKey.....	32
4.12 DeleteAccessKey.....	33
4.13 ListAccessKeys.....	34
4.14 CreateVirtualMFADevice.....	36
4.15 ListVirtualMFADevices.....	38
4.16 DeleteVirtualMFADevice.....	40
4.17 BindMFADevice.....	41
4.18 UnbindMFADevice.....	42
4.19 GetUserMFAInfo.....	44
4.20 ChangePassword.....	45
5 组管理接口.....	47
5.1 CreateGroup.....	47
5.2 GetGroup.....	48
5.3 UpdateGroup.....	50
5.4 ListGroups.....	52
5.5 DeleteGroup.....	54
5.6 AddUserToGroup.....	55
5.7 RemoveUserFromGroup.....	57
5.8 ListGroupsForUser.....	58

5.9 ListUsersForGroup.....	59
6 角色管理接口.....	62
6.1 CreateRole.....	62
6.2 GetRole.....	66
6.3 UpdateRole.....	68
6.4 ListRoles.....	71
6.5 DeleteRole.....	73
7 权限策略管理接口.....	75
7.1 CreatePolicy.....	75
7.2 GetPolicy.....	77
7.3 DeletePolicy.....	79
7.4 ListPolicies.....	81
7.5 CreatePolicyVersion.....	83
7.6 GetPolicyVersion.....	86
7.7 DeletePolicyVersion.....	88
7.8 ListPolicyVersions.....	89
7.9 SetDefaultPolicyVersion.....	91
7.10 AttachPolicyToUser.....	92
7.11 DetachPolicyFromUser.....	94
7.12 AttachPolicyToGroup.....	95
7.13 DetachPolicyFromGroup.....	96
7.14 AttachPolicyToRole.....	98
7.15 DetachPolicyFromRole.....	99
7.16 ListPoliciesForUser.....	100
7.17 ListEntitiesForPolicy.....	102
7.18 ListPoliciesForGroup.....	106
7.19 ListPoliciesForRole.....	108
8 安全设置接口.....	111
8.1 SetAccountAlias.....	111
8.2 GetAccountAlias.....	112
8.3 ClearAccountAlias.....	113
8.4 SetPasswordPolicy.....	114
8.5 GetPasswordPolicy.....	118
8.6 SetSecurityPreference.....	120
9 (隐藏) 旧版 API 参考 (RAM)	124
9.1 简介.....	124
9.1.1 RAM 简介.....	124
9.1.2 术语表.....	125
9.2 错误代码表.....	125
9.3 数据类型.....	137
9.3.1 User.....	138
9.3.2 LoginProfile.....	138
9.3.3 MFADevice.....	139
9.3.4 VirtualMFADevice.....	139

9.3.5 AccessKey.....	140
9.3.6 Group.....	140
9.3.7 Role.....	140
9.3.8 Policy.....	141
9.3.9 PolicyVersion.....	142
9.3.10 PasswordPolicy.....	142
9.3.11 SecurityPreference.....	143
9.4 关于RAM限制.....	144

1 调用方式

1.1 请求结构

本文介绍了RAM的接入地址、通信协议、HTTP请求方法和请求参数等请求结构相关的信息。

接入地址

RAM服务的接入地址为：`https://ram.aliyuncs.com`。

通信协议

为了保证通信的安全性，RAM服务仅支持使用HTTPS安全通道发送请求。

HTTP请求方法

RAM支持HTTP GET/POST方法发送请求。



说明：

GET请求最大不得超过4KB, POST请求最大不得超过10MB。

请求参数

每个请求都需要指定如下信息：

- 要执行的操作：`Action`参数。
- 每个操作接口都需要包含的公共请求参数。
- 操作接口所特有的请求参数。

字符编码

请求及返回结果都使用UTF-8字符集进行编码。

1.2 公共参数

公共参数分为公共请求参数和公共返回参数。

公共请求参数

名称	类型	是否必需	描述
Action	String	是	API的名称。

名称	类型	是否必需	描述
Format	String	否	返回值的类型。支持JSON与XML，默认为XML。
Version	String	是	API版本号。为日期形式：YYYY-MM-DD，本版本对应为2015-05-01。
Signature	String	是	消息签名。
SignatureMethod	String	是	签名方式，目前仅支持HMAC-SHA1。
SignatureNonce	String	是	唯一随机数。用于防止网络重放攻击。用户在不同请求间要使用不同的随机数值。
SignatureVersion	String	是	签名算法版本，目前版本是1.0。
AccessKeyId	String	是	访问密钥ID。
Timestamp	String	是	请求的时间戳。日期格式按照ISO8601标准表示，并需要使用UTC时间。格式为：YYYY-MM-DDThh:mm:ssZ。

公共返回参数

名称	类型	描述
RequestId	String	请求ID。无论成功与否，系统都会返回一个唯一识别码给用户。此参数用于识别每一次请求。

请求示例

```
https://ram.aliyuncs.com/?Action=XXXXXX
?Format=xml
&Version=2015-05-01
&Signature=Pc5WB8gokVn0xfeu%2FZV%2BiNM1dg****
&SignatureMethod=HMAC-SHA1
&SignatureNonce=1521552885****
&SignatureVersion=1.0
&AccessKeyId=key-test
```



```
&Timestamp=2012-06-01T12:00:00Z  
...
```

返回示例

· 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 */  
}
```

1.3 计算签名

RAM服务会对每个访问的请求进行身份验证，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。

背景信息

RAM通过使用AccessKey ID和AccessKeySecret进行对称加密的方法来验证请求的发送者身份。AccessKey ID和AccessKeySecret由阿里云颁发给访问者，其中AccessKey ID用于标识访问者的身份，AccessKeySecret用于加密签名字符串和服务器端验证签名字符串的密钥，必须严格保密谨防泄露。

操作步骤

1. 使用请求参数构造规范化的请求字符串。

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



说明:

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

b) 编码参数。使用UTF-8字符集对每个请求参数的名称和参数取值进行URL编码，URL编码规则如下：

- 对于字符A~Z、a~z、0~9以及字符-、_、.和~不编码。
- 对于其他字符编码成%XY的格式，其中XY是字符对应ASCII码的16进制。例如：半角双引号 (") 对应的编码为：%22。
- 空格 () 需要被编码成：%20，而不是+。



说明：

一般支持URL编码的库都按照application/x-www-form-urlencoded的MIME格式的规则进行编码，例如：Java中的java.net.URLEncoder。您可以直接使用这类方式进行编码，然后将编码后的字符串中加号+替换成%20、*替换成%2A、%7E替换~，即可得到上述规则描述的编码字符串。

c) 使用=连接编码后的参数名称和参数取值。

d) 使用&连接编码后的参数，得到规范化请求字符串。



说明：

参数排序需与**排序参数**保持一致。

2. 使用规范化请求字符串构造用于计算签名的字符串。规则如下：

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

3. 按照RFC2104的定义，计算签名HMAC值。



说明：

计算签名时使用的key就是用户持有的AccessKeySecret加上一个&字符，其ASCII值为38，使用的哈希算法是：SHA1。

4. 按照Base64编码规则将HMAC值编码成字符串，得到签名值。

5. 将签名值作为Signature添加到请求参数中。



说明：

将签名值添加到请求参数时，需要按照RFC3986的规则进行URL编码。

预期结果

下文以CreateUser为例，介绍签名的一个具体示例及结果。

签名前的请求URL为：

```
https://ram.aliyuncs.com/?UserName=test&SignatureVersion=1.0&Format=JSON&Timestamp=2015-08-18T03%3A15%3A45Z&AccessKeyId=testid&SignatureMethod=HMAC-SHA1&Version=2015-05-01&Action=CreateUser&SignatureNonce=6a6e0ca6-4557-11e5-86a2-b8e8563dc8d2
```

对应的StringToSign为：

```
GET&%2F&AccessKeyId%3Dtestid%26Action%3DCreateUser%26Format%3DJSON%26SignatureMethod%3DHMAC-SHA1%26SignatureNonce%3D6a6e0ca6-4557-11e5-86a2-b8e8563dc8d2%26SignatureVersion%3D1.0%26Timestamp%3D2015-08-18T03%253A15%253A45Z%26UserName%3Dtest%26Version%3D2015-05-01
```

例如：AccessKey ID为：testid，AccessKeySecret为：testsecret，则用于计算HMAC的key为：testsecret&。

计算得到的签名值为：kRA2cnpJVacIhDMzXnoNZG9tDCI%3D。

签名后的请求URL为：

```
https://ram.aliyuncs.com/?UserName=test&SignatureVersion=1.0&Format=JSON&Timestamp=2015-08-18T03%3A15%3A45Z&AccessKeyId=testid&SignatureMethod=HMAC-SHA1&Version=2015-05-01&Signature=kRA2cnpJVacIhDMzXnoNZG9tDCI%3D&Action=CreateUser&SignatureNonce=6a6e0ca6-4557-11e5-86a2-b8e8563dc8d2
```

1.4 返回结果

调用RAM API后返回数据采用统一格式，返回结果格式主要有XML和JSON两种，默认为XML格式。本文档中的返回示例为了便于用户查看，做了格式化处理，实际返回结果是没有进行换行、缩进等处理的。

成功结果

调用RAM API后，如果返回的HTTP状态码为：2xx，代表调用成功。

· 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 */
}
```

错误结果

调用STS API后，如果返回的HTTP状态码为：`4xx`或`5xx`，代表调用失败，系统将不会返回结果数据。此时，返回的消息体中包含：具体的错误代码、错误信息、全局唯一的请求ID：`RequestId`以及本次请求访问的站点ID：`HostId`。您可以通过错误码定位问题，详情请参见[查看本产品错误码](#)。

- XML示例

```
<?xml version="1.0" encoding="UTF-8"?>
<Error>
  <RequestId>8906582E-6722-409A-A6C4-0E7863B733A5</RequestId>
  <HostId>ram.aliyuncs.com</HostId>
  <Code>InvalidParameter</Code>
  <Message>The specified parameter "Action or Version" is not valid
.</Message>
</Error>
```

- JSON示例

```
{
  "RequestId": "7463B73D-35CC-4D19-A010-6B8D65D242EF",
  "HostId": "ram.aliyuncs.com",
  "Code": "InvalidParameter",
  "Message": "The specified parameter \"Action or Version\" is not
valid."
}
```

2 RAM鉴权

本文列举了RAM中可授权的API及其描述方式。

当其他账号通过API访问云账号的资源时，RAM需进行权限检查，以确保资源拥有者的确将相关资源的权限授予了调用者。不同的API会根据涉及的资源以及API语义确定需要检查哪些资源的权限。

API的鉴权规则如下表所示：

API	资源描述
ram:CreateUser	acs:ram:*:\${AccountId}:user/*
ram:GetUser	acs:ram:*:\${AccountId}:user/\${UserName}
ram:UpdateUser	acs:ram:*:\${AccountId}:user/\${UserName}
ram>DeleteUser	acs:ram:*:\${AccountId}:user/\${UserName}
ram:ListUsers	acs:ram:*:\${AccountId}:user/*
ram:CreateLoginProfile	acs:ram:*:\${AccountId}:user/\${UserName}
ram:GetLoginProfile	acs:ram:*:\${AccountId}:user/\${UserName}
ram>DeleteLoginProfile	acs:ram:*:\${AccountId}:user/\${UserName}
ram:UpdateLoginProfile	acs:ram:*:\${AccountId}:user/\${UserName}
ram:CreateAccessKey	acs:ram:*:\${AccountId}:user/\${UserName}
ram:UpdateAccessKey	acs:ram:*:\${AccountId}:user/\${UserName}
ram>DeleteAccessKey	acs:ram:*:\${AccountId}:user/\${UserName}
ram:ListAccessKeys	acs:ram:*:\${AccountId}:user/\${UserName}

API	资源描述
ram:CreateVirtualMFADevice	acs:ram:*:\${AccountId}:mfa/*
ram:ListVirtualMFADevices	acs:ram:*:\${AccountId}:mfa/*
ram>DeleteVirtualMFADevice	\${SerialNumber}
ram:BindMFADevice	acs:ram:*:\${AccountId}:user/\${UserName}
ram:UnbindMFADevice	acs:ram:*:\${AccountId}:user/\${UserName}
ram:GetUserMFAInfo	acs:ram:*:\${AccountId}:user/\${UserName}
ram:ChangePassword	acs:ram:*:\${AccountId}:user/\${UserName}
ram:CreateGroup	acs:ram:*:\${AccountId}:group/*
ram:GetGroup	acs:ram:*:\${AccountId}:group/\${GroupName}
ram:UpdateGroup	acs:ram:*:\${AccountId}:group/\${GroupName}
ram:ListGroups	acs:ram:*:\${AccountId}:group/*
ram>DeleteGroup	acs:ram:*:\${AccountId}:group/\${GroupName}
ram:AddUserToGroup	acs:ram:*:\${AccountId}:user/\${UserName}
	acs:ram:*:\${AccountId}:group/\${GroupName}
ram:RemoveUserFromGroup	acs:ram:*:\${AccountId}:user/\${UserName}
	acs:ram:*:\${AccountId}:group/\${GroupName}
ram:ListGroupsForUser	acs:ram:*:\${AccountId}:user/\${UserName}
ram:ListUsersForGroup	acs:ram:*:\${AccountId}:group/\${GroupName}
ram:CreateRole	acs:ram:*:\${AccountId}:role/*

API	资源描述
ram:GetRole	acs:ram:*:\${AccountId}:role/\${RoleName}
ram:UpdateRole	acs:ram:*:\${AccountId}:role/\${RoleName}
ram:ListRoles	acs:ram:*:\${AccountId}:role/*
ram>DeleteRole	acs:ram:*:\${AccountId}:role/\${RoleName}
ram>CreatePolicy	acs:ram:*:\${AccountId}:policy/*
ram:GetPolicy	acs:ram:*:\${AccountId} or system:policy/\${PolicyName}
ram>DeletePolicy	acs:ram:*:\${AccountId}:policy/\${PolicyName}
ram:ListPolicies	acs:ram:*:\${AccountId}:policy/*
ram>CreatePolicyVersion	acs:ram:*:\${AccountId}:policy/\${PolicyName}
ram:GetPolicyVersion	acs:ram:*:\${AccountId} or system:group/\${PolicyName}
ram>DeletePolicyVersion	acs:ram:*:\${AccountId}:policy/\${PolicyName}
ram:ListPolicyVersions	acs:ram:*:\${AccountId} or system:policy/\${PolicyName}
ram:SetDefaultPolicyVersion	acs:ram:*:\${AccountId}:policy/\${PolicyName}
ram:AttachPolicyToUser	acs:ram:*:\${AccountId}:user/\${UserName}
	acs:ram:*:\${AccountId} or system:policy/\${PolicyName}
ram:DetachPolicyFromUser	acs:ram:*:\${AccountId}:user/\${UserName}
	acs:ram:*:\${AccountId} or system:policy/\${PolicyName}

API	资源描述
ram:AttachPolicyToGroup	acs:ram:*:\${AccountId}:group/\${GroupName}
	acs:ram:*:\${AccountId} or system:policy/\${PolicyName}
ram:DetachPolicyFromGroup	acs:ram:*:\${AccountId}:group/\${GroupName}
	acs:ram:*:\${AccountId} or system:policy/\${PolicyName}
ram:AttachPolicyToRole	acs:ram:*:\${AccountId}:role/\${RoleName}
	acs:ram:*:\${AccountId} or system:policy/\${PolicyName}
ram:DetachPolicyFromRole	acs:ram:*:\${AccountId}:role/\${RoleName}
	acs:ram:*:\${AccountId} or system:policy/\${PolicyName}
ram:ListEntitiesForPolicy	acs:ram:*:\${AccountId} or system:policy/\${PolicyName}
ram:SetAccountAlias	acs:ram:*:\${AccountId}:*
ram:GetAccountAlias	acs:ram:*:\${AccountId}:*
ram:ClearAccountAlias	acs:ram:*:\${AccountId}:*
ram:SetPasswordPolicy	acs:ram:*:\${AccountId}:*
ram:GetPasswordPolicy	acs:ram:*:\${AccountId}:*
ram:SetSecurityPreference	acs:ram:*:\${AccountId}:*

3 API概览

本文列出了访问控制（RAM）所有可调用的API接口及相关描述。

调试

前往[API Explorer](#)在线调试，API Explorer提供在线调用API、动态生成SDK Example代码和快速检索接口等能力，能显著降低使用云API的难度，强烈推荐使用。

用户管理相关接口

API	描述
#unique_10	创建RAM用户。
#unique_11	获取用户的详细信息。
#unique_12	更新用户的基本信息。
#unique_13	删除一个RAM用户。
#unique_14	列出所有RAM用户。
#unique_15	为一个RAM用户启用Web控制台登录。
#unique_16	查看一个RAM用户的登录配置。
#unique_17	关闭指定RAM用户登录Web控制台的功能。
#unique_18	修改用户的登录配置。
#unique_19	为RAM用户创建AccessKey。
#unique_20	变更RAM用户AccessKey的状态。
#unique_21	删除RAM用户的AccessKey。
#unique_22	列出指定用户的AccessKeys。
#unique_23	创建虚拟多因素认证设备。
#unique_24	列出虚拟多因素认证设备。
#unique_25	删除虚拟多因素认证设备。
#unique_26	绑定多因素认证设备。
#unique_27	解绑多因素认证设备。
#unique_28	获取指定RAM用户的多因素认证设备信息。
#unique_29	修改RAM用户密码。

组管理相关接口

API	描述
#unique_30	创建用户组。
#unique_31	获取用户组信息。
#unique_32	更新用户组信息。
#unique_33	列出所有的用户组。
#unique_34	删除指定的组。
#unique_35	将RAM用户添加到指定的用户组。
#unique_36	将RAM用户从用户组中移除。
#unique_37	列出指定RAM用户所加入的组信息。
#unique_38	列出指定用户组所包含的RAM用户。

角色管理相关接口

API	描述
#unique_39	创建角色。
#unique_40	获取角色信息。
#unique_41	更新角色信息。
#unique_42	列出角色。
#unique_43	删除指定的角色。

权限策略管理接口

API	描述
#unique_44	创建一个权限策略。
#unique_45	获取指定的权限策略信息。
#unique_46	删除指定的权限策略。
#unique_47	列出权限策略。
#unique_48	为权限策略创建新的版本。
#unique_49	获取某个权限策略的版本。
#unique_50	删除指定的权限策略的某个版本。
#unique_51	列出权限策略版本。
#unique_52	设置权限策略默认版本。

API	描述
#unique_53	为指定用户附加权限。
#unique_54	为用户撤销指定的权限。
#unique_55	为指定组附加权限。
#unique_56	为组撤销指定的权限。
#unique_57	为指定角色附加权限。
#unique_58	为角色撤销指定的权限。
#unique_59	列出引用权限策略的实体。
#unique_60	列出指定用户被授予的权限策略。
#unique_61	列出组被授予的权限策略。
#unique_62	列出角色被授予的权限策略。

安全设置相关接口

API	描述
#unique_63	设置云账号别名。
#unique_64	查看云账号别名。
#unique_65	清除云账号别名。
#unique_66	设置RAM用户密码强度等策略。
#unique_67	获取RAM用户密码强度等策略。
#unique_68	设置全局安全首选项。

4 用户管理接口

4.1 CreateUser

调用CreateUser接口创建RAM用户。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	CreateUser	系统规定参数。取值：CreateUser
Comments	String	否	This is a cloud computing engineer.	备注，最大长度128个字符。
DisplayName	String	否	zhangq****	显示名称，最多包含128个字符或汉字。 格式： <code>^[a-zA-Z0-9\.\@\-\u4e00-\u9fa5]+\$</code> 。
Email	String	否	zhangq****@example.com	RAM用户的邮箱。
MobilePhone	String	否	86-1868888****	RAM用户手机号。 格式：国际区号-号码。
UserName	String	否	zhangq****	指定用户名，最多包含64个字符。 格式： <code>^[a-zA-Z0-9\.\@\-_]+\$</code> 。

返回数据

名称	类型	示例值	描述
RequestId	String	04F0F334-1335-436C-A1D7-6C044FE73368	请求ID。
User			用户信息。
Comments	String	这是一位云计算工程师	备注。
CreateDate	String	2015-01-23T12:33:18Z	创建时间。
DisplayName	String	张强	显示名称。
Email	String	zhangq****@example.com	RAM用户的邮箱。
MobilePhone	String	86-1860000****	RAM用户手机号。
UserId	String	122748924538****	用户唯一标识。
UserName	String	zhangq****	用户名称。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=CreateUser
&UserName=zhangq****
&DisplayName=zhangq****
&MobilePhone=86-1868888****
&Email=zhangq****@example.com
&Comments=This is a cloud computing engineer.
&<公共请求参数>
```

正常返回示例

XML 格式

```
<CreateUserResponse>
  <RequestId>04F0F334-1335-436C-A1D7-6C044FE73368</RequestId>
  <User>
    <UserId>122748924538****</UserId>
    <UserName>zhangq****</UserName>
    <DisplayName>zhangq****</DisplayName>
    <MobilePhone>86-1860000****</MobilePhone>
```

```
<Email>zhangq****@example.com</Email>
<Comments>这是一位云计算工程师</Comments>
<CreateDate>2015-01-23T12:33:18Z</CreateDate>
</User>
</CreateUserResponse>
```

JSON 格式

```
{
  "User":{
    "Comments":"这是一位云计算工程师",
    "Email":"zhangq****@example.com",
    "UserName":"zhangq****",
    "UserId":"122748924538****",
    "MobilePhone":"86-1860000****",
    "Displayname":"zhangq****",
    "CreateDate":"2015-01-23T12:33:18Z"
  },
  "RequestId":"04F0F334-1335-436C-A1D7-6C044FE73368"
}
```

错误码

访问[错误中心](#)查看更多错误码。

4.2 GetUser

调用GetUser接口获取用户的详细信息。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	GetUser	系统规定参数。取值：GetUser
UserName	String	是	zhangq****	指定用户名。 格式： <code>^[a-zA-Z0-9\.\@\-_]+\$</code> 。

返回数据

名称	类型	示例值	描述
RequestId	String	2D69A58F-345C-4FDE-88E4-BF5189484043	请求ID。

名称	类型	示例值	描述
User			用户信息。
Comments	String	这是一位云计算工程师	备注。
CreateDate	String	2015-01-23T12:33:18Z	创建时间。
DisplayName	String	张*	显示名称。
Email	String	zhangq****@example.com	RAM用户的邮箱。
LastLoginDate	String	2015-01-23T12:33:18Z	上次使用密码登录时间。
MobilePhone	String	86-1860000****	RAM用户手机号。
UpdateDate	String	2015-02-11T03:15:21Z	更新时间。
UserId	String	122748924538****	用户唯一标识。
UserName	String	zhangq****	用户名称。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=GetUser
&UserName=zhangq****
&<公共请求参数>
```

正常返回示例

XML 格式

```
<GetUserResponse>
  <RequestId>2D69A58F-345C-4FDE-88E4-BF5189484043</RequestId>
  <User>
    <UserId>122748924538****</UserId>
    <UserName>zhangq****</UserName>
    <DisplayName>张*</DisplayName>
    <MobilePhone>86-1860000****</MobilePhone>
    <Email>zhangq****@example.com</Email>
    <Comments>这是一位云计算工程师</Comments>
    <CreateDate>2015-01-23T12:33:18Z</CreateDate>
    <UpdateDate>2015-02-11T03:15:21Z</UpdateDate>
    <LastLoginDate>2015-01-23T12:33:18Z</LastLoginDate>
```

```
</User>
</GetUserResponse>
```

JSON 格式

```
{
  "User":{
    "LastLoginDate":"2015-01-23T12:33:18Z",
    "Comments":"这是一位云计算工程师",
    "Email":"zhangq****@example.com",
    "UserName":"zhangq****",
    "UpdateDate":"2015-02-11T03:15:21Z",
    "UserId":"122748924538****",
    "MobilePhone":"86-1860000****",
    "DisplayName":"张*",
    "CreateDate":"2015-01-23T12:33:18Z"
  },
  "RequestId":"2D69A58F-345C-4FDE-88E4-BF5189484043"
}
```

错误码

访问[错误中心](#)查看更多错误码。

4.3 UpdateUser

调用UpdateUser接口更新用户的基本信息。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	UpdateUser	系统规定参数。取值：UpdateUser
UserName	String	否	zhangq****	指定用户名。
NewUserName	String	否	xiaoq****	指定新用户名。 格式： <code>^[a-zA-Z0-9\.\@_\-]+\$。</code>
NewMobilePhone	String	否	86-1860000****	指定RAM用户新手机号。 格式：国际区号-号码。
NewDisplayName	String	否	xiaoq****	指定新显示名称。

名称	类型	是否必选	示例值	描述
NewEmail	String	否	xiaoq****@example.com	指定RAM用户的新邮箱。
NewComments	String	否	这是一位云计算工程师	指定新备注, 最大长度128个字符。

返回数据

名称	类型	示例值	描述
RequestId	String	04F0F334-1335-436C-A1D7-6C044FE73368	请求ID。
User			用户信息。
Comments	String	这是一位云计算工程师	备注。
CreateDate	String	2015-01-23T12:33:18Z	创建时间。
DisplayName	String	xiaoq****	显示名称。
Email	String	xiaoq****@example.com	RAM用户邮箱。
MobilePhone	String	86-1860000****	RAM用户手机号。
UpdateDate	String	2015-02-11T03:15:21Z	更新时间。
UserId	String	122748924538****	用户唯一标识。
UserName	String	xiaoq****	用户名称。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=UpdateUser
&UserName=zhangq****
&NewUserName=xiaoq****
&NewMobilePhone=86-1860000****
&NewEmail=xiaoq****@example.com
```

&<公共请求参数>

正常返回示例

XML 格式

```
<CreateUserResponse>
  <RequestId>04F0F334-1335-436C-A1D7-6C044FE73368</RequestId>
  <User>
    <UserId>122748924538****</UserId>
    <UserName>xiaoq****</UserName>
    <DisplayName>xiaoq****</DisplayName>
    <MobilePhone>86-1860000****</MobilePhone>
    <Email>xiaoq****@example.com</Email>
    <Comments>这是一位云计算工程师</Comments>
    <CreateDate>2015-01-23T12:33:18Z</CreateDate>
    <UpdateDate>2015-02-11T03:15:21Z</UpdateDate>
  </User>
</CreateUserResponse>
```

JSON 格式

```
{
  "User": {
    "Comments": "这是一位云计算工程师",
    "Email": "xiaoq****@example.com",
    "UserName": "xiaoq****",
    "UpdateDate": "2015-02-11T03:15:21Z",
    "UserId": "122748924538****",
    "MobilePhone": "86-1860000****",
    "DisplayName": "xiaoq****",
    "CreateDate": "2015-01-23T12:33:18Z"
  },
  "RequestId": "04F0F334-1335-436C-A1D7-6C044FE73368"
}
```

错误码

访问[错误中心](#)查看更多错误码。

4.4 DeleteUser

调用DeleteUser接口删除一个RAM用户。

删除用户前，需要保证用户不拥有任何权限策略(Policy)，并且不属于任何用户组。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	DeleteUser	系统规定参数。取值: DeleteUser
UserName	String	是	zhangq****	指定用户名。 格式: <code>^[a-zA-Z0-9\.\@_\-]+\$</code> 。

返回数据

名称	类型	示例值	描述
RequestId	String	1C488B66-B819-4D14-8711-C4EAAA13AC01	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=DeleteUser
&UserName=zhangq****
&<公共请求参数>
```

正常返回示例

XML 格式

```
<DeleteUserResponse>
  <RequestId>1C488B66-B819-4D14-8711-C4EAAA13AC01</RequestId>
</DeleteUserResponse>
```

JSON 格式

```
{
  "RequestId": "1C488B66-B819-4D14-8711-C4EAAA13AC01"
}
```

错误码

访问[错误中心](#)查看更多错误码。

4.5 ListUsers

调用ListUsers接口列出所有RAM用户。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	ListUsers	系统规定参数。取值：ListUsers
Marker	String	否	EXAMPLE	当请求的返回结果被截断时，可以使用Marker获取从当前截断位置之后的内容。
MaxItems	Integer	否	100	指定返回结果的条数，当返回结果达到MaxItems限制被截断时，返回参数IsTruncated将等于true。 取值范围：1 ~ 100，默认值：100。

返回数据

名称	类型	示例值	描述
IsTruncated	Boolean	true	请求返回结果是否被截断。
Marker	String	EXAMPLE	当IsTruncated为true时才有此字段，当返回true时，需要继续调用此接口，并且使用Marker获取截断后的内容。
RequestId	String	4B450CA1-36E8-4AA2-8461-86B42BF4CC4E	请求ID。
Users			用户信息集合。

名称	类型	示例值	描述
Comments	String	这是一位云计算工程师	备注。
CreateDate	String	2015-01-23T12:33:18Z	创建时间。
DisplayName	String	张*	显示名称。
Email	String	zhangq****@example.com	RAM用户邮箱。
MobilePhone	String	86-1860000****	RAM用户手机号。
UpdateDate	String	2015-01-23T12:33:18Z	更新时间。
UserId	String	122748924538****	用户唯一标识。
UserName	String	zhangq****	用户名称。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=ListUsers
&<公共请求参数>
```

正常返回示例

XML 格式

```
<ListUsersResponse>
  <RequestId>4B450CA1-36E8-4AA2-8461-86B42BF4CC4E</RequestId>
  <Users>
    <User>
      <UserId>122748924538****</UserId>
      <UserName>zhangq****</UserName>
      <DisplayName>张*</DisplayName>
      <MobilePhone>86-1860000****</MobilePhone>
      <Email>zhangq****@example.com</Email>
      <Comments>这是一位云计算工程师</Comments>
      <CreateDate>2015-01-23T12:33:18Z</CreateDate>
      <UpdateDate>2015-01-23T12:33:18Z</UpdateDate>
    </User>
    <User>
      <UserId>140649822472****</UserId>
      <UserName>li****</UserName>
      <DisplayName>李*</DisplayName>
      <MobilePhone>86-1860000****</MobilePhone>
      <Email>li****@example.com</Email>
```

```
<Comments>权限管理员</Comments>
<CreateDate>2015-02-18T17:22:08Z</CreateDate>
<UpdateDate>2015-02-18T17:22:08Z</UpdateDate>
</User>
</Users>
<IsTruncated>true</IsTruncated>
<Marker>EXAMPLE</Marker>
</ListUsersResponse>
```

JSON 格式

```
{
  "Users":{
    "User":[
      {
        "Comments":"这是一位云计算工程师",
        "Email":"zhangq***@example.com",
        "UserName":"zhangq***",
        "UpdateDate":"2015-01-23T12:33:18Z",
        "UserId":"122748924538****",
        "MobilePhone":"86-1860000****",
        "DisplayName":"张*",
        "CreateDate":"2015-01-23T12:33:18Z"
      },
      {
        "Comments":"权限管理员",
        "Email":"li***@example.com",
        "UserName":"li***",
        "UpdateDate":"2015-02-18T17:22:08Z",
        "UserId":"140649822472****",
        "MobilePhone":"86-1860000****",
        "DisplayName":"李*",
        "CreateDate":"2015-02-18T17:22:08Z"
      }
    ]
  },
  "IsTruncated":true,
  "RequestId":"4B450CA1-36E8-4AA2-8461-86B42BF4CC4E",
  "Marker":"EXAMPLE"
}
```

错误码

访问[错误中心](#)查看更多错误码。

4.6 CreateLoginProfile

调用CreateLoginProfile接口为一个RAM用户启用Web控制台登录。

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。

请求参数

名称	类型	是否必选	示例值	描述
UserName	String	是	zhangq****	指定用户名。
Password	String	是	mypassword	指定密码，密码必须符合密码强度要求。关于密码强度设置接口，请参考 GetPasswordPolicy 。
Action	String	是	CreateLoginProfile	系统规定参数。取值：CreateLoginProfile
PasswordResetRequired	Boolean	否	false	指定用户在登录时是否需要修改密码。默认为：false。
MFABindRequired	Boolean	否	false	指定用户在下次登录时是否必须绑定多因素认证器。默认为：false。

返回数据

名称	类型	示例值	描述
LoginProfile			登录配置信息。
CreateDate	String	2015-01-23T12:33:18Z	创建时间。
MFABindRequired	Boolean	false	要求必须绑定多因素认证设备。
PasswordResetRequired	Boolean	false	要求下次登录时重设密码。
UserName	String	zhangq****	用户名。
RequestId	String	04F0F334-1335-436C-A1D7-6C044FE73368	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=CreateLoginProfile
```

```
&UserName=zhangq****
&Password=mypassword
&PasswordResetRequired=false
&<公共请求参数>
```

正常返回示例

XML 格式

```
<CreateLoginProfile>
  <RequestId>04F0F334-1335-436C-A1D7-6C044FE73368</RequestId>
  <LoginProfile>
    <UserName>zhangq****</UserName>
    <PasswordResetRequired>>true</PasswordResetRequired>
    <MFABindRequired>>true</MFABindRequired>
    <CreateDate>2015-01-23T12:33:18Z</CreateDate>
  </LoginProfile>
</CreateLoginProfile>
```

JSON 格式

```
{
  "RequestId": "04F0F334-1335-436C-A1D7-6C044FE73368",
  "LoginProfile": {
    "PasswordResetRequired": true,
    "MFABindRequired": true,
    "UserName": "zhangq****",
    "CreateDate": "2015-01-23T12:33:18Z"
  }
}
```

错误码

访问[错误中心](#)查看更多错误码。

4.7 GetLoginProfile

调用GetLoginProfile接口查看一个RAM用户的登录配置。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	GetLoginProfile	系统规定参数。取值：GetLoginProfile
UserName	String	是	zhangq****	指定用户名。

返回数据

名称	类型	示例值	描述
LoginProfile			登录配置信息。
CreateDate	String	2015-01-23T12:33:18Z	创建时间。
MFABindRequired	Boolean	true	要求必须绑定多因素认证设备。
PasswordResetRequired	Boolean	true	要求下次登录时重设密码。
UserName	String	zhangq****	用户名。
RequestId	String	04F0F334-1335-436C-A1D7-6C044FE73368	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=GetLoginProfile
&UserName=zhangq****
&<公共请求参数>
```

正常返回示例

XML 格式

```
<GetLoginProfile>
  <RequestId>04F0F334-1335-436C-A1D7-6C044FE73368</RequestId>
  <LoginProfile>
    <UserName>zhangq****</UserName>
    <PasswordResetRequired>true</PasswordResetRequired>
    <MFABindRequired>true</MFABindRequired>
    <CreateDate>2015-01-23T12:33:18Z</CreateDate>
  </LoginProfile>
</GetLoginProfile>
```

JSON 格式

```
{
  "RequestId": "04F0F334-1335-436C-A1D7-6C044FE73368",
  "LoginProfile": {
    "PasswordResetRequired": true,
    "MFABindRequired": true,
    "UserName": "zhangq****",
    "CreateDate": "2015-01-23T12:33:18Z"
  }
}
```

```
}
```

错误码

访问[错误中心](#)查看更多错误码。

4.8 DeleteLoginProfile

调用DeleteLoginProfile接口关闭指定RAM用户登录Web控制台的功能。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	DeleteLoginProfile	系统规定参数。取值：DeleteLoginProfile
UserName	String	是	zhangq****	指定用户名。

返回数据

名称	类型	示例值	描述
RequestId	String	1C488B66-B819-4D14-8711-C4EAAA13AC01	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=DeleteLoginProfile
&UserName=zhangq****
&<公共请求参数>
```

正常返回示例

XML 格式

```
<DeleteLoginProfileResponse>
  <RequestId>1C488B66-B819-4D14-8711-C4EAAA13AC01</RequestId>
```

```
</DeleteLoginProfileResponse>
```

JSON 格式

```
{
  "RequestId": "1C488B66-B819-4D14-8711-C4EAAA13AC01"
}
```

错误码

访问[错误中心](#)查看更多错误码。

4.9 UpdateLoginProfile

调用UpdateLoginProfile接口修改用户的登录配置。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	UpdateLoginProfile	系统规定参数。取值：UpdateLoginProfile
UserName	String	是	zhangq****	指定用户名。
MFABindRequired	Boolean	否	true	指定用户在下次登录时是否必须绑定多因素认证设备。
Password	String	否	mypassword	指定密码，密码必须符合主账号的密码强度要求。
PasswordResetRequired	Boolean	否	true	指定用户在登录时是否需要修改密码。

返回数据

名称	类型	示例值	描述
RequestId	String	04F0F334-1335-436C-A1D7-6C044FE73368	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=UpdateLoginProfile
&UserName=zhangq****
&Password=mypassword
&PasswordResetRequired=true
&<公共请求参数>
```

正常返回示例

XML 格式

```
<UpdateLoginProfile>
  <RequestId>04F0F334-1335-436C-A1D7-6C044FE73368</RequestId>
</UpdateLoginProfile>
```

JSON 格式

```
{
  "RequestId": "04F0F334-1335-436C-A1D7-6C044FE73368"
}
```

错误码

访问[错误中心](#)查看更多错误码。

4.10 CreateAccessKey

调用CreateAccessKey接口为RAM用户创建访问密钥。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	CreateAccessKey	系统规定参数。取值：CreateAccessKey
UserName	String	否	zhangq****	指定用户名，RAM用户调用此接口时，默认为自己创建访问密钥。

返回数据

名称	类型	示例值	描述
AccessKey			访问密钥。
AccessKeyId	String	0wNEpMMLzy7s****	访问密钥标识。
AccessKeySecret	String	PupkTg8jdm au1cXxYacg E736PJ****	访问密钥。
CreateDate	String	2015-01-23T12:33:18Z	创建时间。
Status	String	Active	状态，激活或禁用。
RequestId	String	04F0F334-1335-436C-A1D7-6C044FE73368	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=CreateAccessKey
&UserName=zhangq****
&<公共请求参数>
```

正常返回示例

XML 格式

```
<CreateAccessKeyResponse>
  <RequestId>04F0F334-1335-436C-A1D7-6C044FE73368</RequestId>
  <AccessKey>
    <AccessKeyId>0wNEpMMLzy7s****</AccessKeyId>
    <AccessKeySecret>PupkTg8jdm  
au1cXxYacgE736PJ****</
AccessKeySecret>
    <Status>Active</Status>
    <CreateDate>2015-01-23T12:33:18Z</CreateDate>
  </AccessKey>
</CreateAccessKeyResponse>
```

JSON 格式

```
{
  "AccessKey":{
    "Status":"Active",
    "AccessKeySecret":"PupkTg8jdm  
au1cXxYacgE736PJ****",
    "AccessKeyId":"0wNEpMMLzy7s****",
    "CreateDate":"2015-01-23T12:33:18Z"
```

```

    },
    "RequestId": "04F0F334-1335-436C-A1D7-6C044FE73368"
  }
}

```

错误码

访问[错误中心](#)查看更多错误码。

4.11 UpdateAccessKey

调用UpdateAccessKey接口更新RAM用户访问密钥的状态。

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。

请求参数

名称	类型	是否必选	示例值	描述
UserAccessKeyId	String	是	0wNEpMMlzy7s****	指定要更新的AccessKeyId。
Status	String	是	Inactive	AccessKey的状态，取值为：Active或Inactive。
Action	String	是	UpdateAccessKey	系统规定参数。取值：UpdateAccessKey。
UserName	String	否	zhangq****	指定用户名。

返回数据

名称	类型	示例值	描述
RequestId	String	04F0F334-1335-436C-A1D7-6C044FE73368	请求ID。

示例

请求示例

```

https://ram.aliyuncs.com/?Action=UpdateAccessKey
&UserName=zhangq****
&UserAccessKeyId=0wNEpMMlzy7s****
&Status=Inactive

```

<公共请求参数>

正常返回示例

XML 格式

```
<UpdateAccessKeyResponse>
  <RequestId>04F0F334-1335-436C-A1D7-6C044FE73368</RequestId>
</UpdateAccessKeyResponse>
```

JSON 格式

```
{
  "RequestId": "04F0F334-1335-436C-A1D7-6C044FE73368"
}
```

错误码

访问[错误中心](#)查看更多错误码。

4.12 DeleteAccessKey

调用DeleteAccessKey接口删除RAM用户的访问密钥。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	DeleteAccessKey	系统规定参数。取值：DeleteAccessKey
UserAccessKeyId	String	是	0wNEpMMlzy7s****	指定要删除的AccessKeyId。
UserName	String	否	zhangq****	指定用户名。

返回数据

名称	类型	示例值	描述
RequestId	String	04F0F334-1335-436C-A1D7-6C044FE73368	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=DeleteAccessKey
&UserName=zhangq****
&UserAccessKeyId=0wNEpMMlzy7s****
&<公共请求参数>
```

正常返回示例

XML 格式

```
<DeleteAccessKeyResponse>
  <RequestId>04F0F334-1335-436C-A1D7-6C044FE73368</RequestId>
</DeleteAccessKeyResponse>
```

JSON 格式

```
{
  "RequestId": "04F0F334-1335-436C-A1D7-6C044FE73368"
}
```

错误码

访问[错误中心](#)查看更多错误码。

4.13 ListAccessKeys

调用ListAccessKeys接口列出指定用户的访问密钥。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	ListAccessKeys	系统规定参数。取值：ListAccessKeys
UserName	String	否	zhangq****	指定用户，RAM用户访问时不提供此参数则表示列出自己的访问密钥。

返回数据

名称	类型	示例值	描述
AccessKeys			用户信息集合。
AccessKeyId	String	0wNEpMMLzy7s****	访问密钥标识。
CreateDate	String	2015-01-23T12:33:18Z	创建时间。
Status	String	Active	状态，激活或禁用。
RequestId	String	4B450CA1-36E8-4AA2-8461-86B42BF4CC4E	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=ListAccessKeys
&<公共请求参数>
```

正常返回示例

XML 格式

```
<ListAccessKeysResponse>
  <RequestId>4B450CA1-36E8-4AA2-8461-86B42BF4CC4E</RequestId>
  <AccessKeys>
    <AccessKey>
      <AccessKeyId>0wNEpMMLzy7s****</AccessKeyId>
      <Status>Active</Status>
      <CreateDate>2015-01-23T12:33:18Z</CreateDate>
    </AccessKey>
    <AccessKey>
      <AccessKeyId>WnIWUruvfaDT****</AccessKeyId>
      <Status>Inactive</Status>
      <CreateDate>2015-03-24T21:12:21Z</CreateDate>
    </AccessKey>
  </AccessKeys>
</ListAccessKeysResponse>
```

JSON 格式

```
{
  "RequestId": "4B450CA1-36E8-4AA2-8461-86B42BF4CC4E",
  "AccessKeys": {
    "AccessKey": [
      {
        "Status": "Active",
        "AccessKeyId": "0wNEpMMLzy7s****",
```

```

    "CreateDate": "2015-01-23T12:33:18Z"
  },
  {
    "Status": "Inactive",
    "AccessKeyId": "WnIWUruvfaDT****",
    "CreateDate": "2015-03-24T21:12:21Z"
  }
]
}
}
}

```

错误码

访问[错误中心](#)查看更多错误码。

4.14 CreateVirtualMFADevice

调用CreateVirtualMFADevice接口创建多因素认证设备。

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。

请求参数

名称	类型	是否必选	示例值	描述
VirtualMFA DeviceName	String	是	device001	指定设备名称, 最大长度64个字符。 限制: [a-zA-Z0-9-]*。

返回数据

名称	类型	示例值	描述
RequestId	String	04F0F334-1335 -436C-A1D7- 6C044FE73368	请求ID。
VirtualMFA Device			多因素认证设备。
Base32Stri ngSeed	String	DSF98HAD98 2KJA9SDFNA S9D8FU839B 8ADHBGS****	多因素认证设备密钥。

名称	类型	示例值	描述
QRCodePNG	String	YXNkZmFzZD lmeW5hc2Q5 OGZoODd4bX JmcThhaGU5 aSBmYXNkZi BzYWWRmIGFG IDRxd2VjIG EgdHEz****	密钥二维码PNG，使用Base64编码。
SerialNumber	String	acs:ram:: 1234567890 12****:mfa/ device001	设备序列号。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=CreateVirtualMFADevice
&VirtualMFADeviceName=device001
&<公共请求参数>
```

正常返回示例

XML 格式

```
<CreateVirtualMFADeviceResponse>
  <RequestId>04F0F334-1335-436C-A1D7-6C044FE73368</RequestId>
  <VirtualMFADevice>
    <SerialNumber>acs:ram::123456789012****:mfa/device001</
SerialNumber>
    <Base32StringSeed>DSF98HAD982KJA9SDFNAS9D8FU839B8ADHBGS
****</Base32StringSeed>
    <QRCodePNG>YXNkZmFzZDlmeW5hc2Q5OGZoODd4bXJmcThhaGU5
aSBmYXNkZiBzYWWRmIGFGIDRxd2VjIGEgdHEz****</QRCodePNG>
  </VirtualMFADevice>
</CreateVirtualMFADeviceResponse>
```

JSON 格式

```
{
  "VirtualMFADevice":{
    "SerialNumber":"acs:ram::123456789012****:mfa/device001",
    "QRCodePNG":"YXNkZmFzZDlmeW5hc2Q5OGZoODd4bXJmcThhaGU5aSBmYXNkZi
BzYWWRmIGFGIDRxd2VjIGEgdHEz****",
    "Base32StringSeed":"DSF98HAD982KJA9SDFNAS9D8FU839B8ADHBGS****"
  },
  "RequestId":"04F0F334-1335-436C-A1D7-6C044FE73368"
```

```
}

```

错误码

访问[错误中心](#)查看更多错误码。

4.15 ListVirtualMFADevices

调用ListVirtualMFADevices接口列出多因素认证设备。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	ListVirtualMFADevices	系统规定参数。取值：ListVirtualMFADevices

返回数据

名称	类型	示例值	描述
RequestId	String	04F0F334-1335-436C-A1D7-6C044FE73368	请求ID。
VirtualMFADevices			多因素认证设备列表。
ActivateDate	String	2015-02-18T17:22:08Z	激活日期。
SerialNumber	String	acs:ram::123456789012****:mfa/device002	多因素认证设备序列号。
User			绑定用户的基本信息。
DisplayName	String	张*	显示名称。
UserId	String	122748924538****	用户唯一标识。

名称	类型	示例值	描述
UserName	String	zhangq****	用户名称。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=ListVirtualMFADevices
&<公共请求参数>
```

正常返回示例

XML 格式

```
<ListVirtualMFADevicesResponse>
  <RequestId>04F0F334-1335-436C-A1D7-6C044FE73368</RequestId>
  <VirtualMFADevices>
    <VirtualMFADevice>
      <SerialNumber>acs:ram::123456789012****:mfa/
device001</SerialNumber>
    </VirtualMFADevice>
    <VirtualMFADevice>
      <SerialNumber>acs:ram::123456789012****:mfa/
device002</SerialNumber>
      <ActivateDate>2015-02-18T17:22:08Z</ActivateDate>
      <User>
        <UserId>122748924538****</UserId>
        <UserName>zhangq****</UserName>
        <DisplayName>张*</DisplayName>
      </User>
    </VirtualMFADevice>
  </VirtualMFADevices>
</ListVirtualMFADevicesResponse>
```

JSON 格式

```
{
  "RequestId": "04F0F334-1335-436C-A1D7-6C044FE73368",
  "VirtualMFADevices": {
    "VirtualMFADevice": [
      {
        "SerialNumber": "acs:ram::123456789012****:mfa/device001"
      },
      {
        "User": {
          "UserName": "zhangq****",
          "UserId": "122748924538****",
          "DisplayName": "张*"
        },
        "SerialNumber": "acs:ram::123456789012****:mfa/device002",
        "ActivateDate": "2015-02-18T17:22:08Z"
      }
    ]
  }
}
```

```
}

```

错误码

访问[错误中心](#)查看更多错误码。

4.16 DeleteVirtualMFADevice

调用DeleteVirtualMFADevice接口删除多因素认证设备。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	DeleteVirtualMFADevice	系统规定参数。取值：DeleteVirtualMFADevice
SerialNumber	String	是	acs:ram:: 1234567890 12****:mfa/ device002	指定多因素认证设备的序列号。

返回数据

名称	类型	示例值	描述
RequestId	String	04F0F334-1335 -436C-A1D7- 6C044FE73368	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=DeleteVirtualMFADevice
&SerialNumber=acs:ram::  
123456789012****:mfa/  
device002
&<公共请求参数>
```

正常返回示例

XML 格式

```
<DeleteVirtualMFADeviceResponse>
```

```
<RequestId>04F0F334-1335-436C-A1D7-6C044FE73368</RequestId>
</DeleteVirtualMFADeviceResponse>
```

JSON 格式

```
{
  "RequestId": "04F0F334-1335-436C-A1D7-6C044FE73368"
}
```

错误码

访问[错误中心](#)查看更多错误码。

4.17 BindMFADevice

调用BindMFADevice接口绑定多因素认证设备。

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。

请求参数

名称	类型	是否必选	示例值	描述
SerialNumber	String	是	acs:ram:: 1234567890 12****:mfa/ device002	指定多因素认证设备的序列号。
UserName	String	是	zhangq****	指定用户名。
AuthenticationCode1	String	是	11****	验证第一组动态密码。
AuthenticationCode2	String	是	33****	验证第二组动态密码。
Action	String	是	BindMFADev ice	系统规定参数。取值： BindMFADevice

返回数据

名称	类型	示例值	描述
RequestId	String	04F0F334-1335 -436C-A1D7- 6C044FE73368	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=BindMFADevice
&SerialNumber=acs:ram::123456789012****:mfa/device002
&UserName=zhangq****
&AuthenticationCode1=11****
&AuthenticationCode2=33****
&<公共请求参数>
```

正常返回示例

XML 格式

```
<BindMFADeviceResponse>
  <RequestId>04F0F334-1335-436C-A1D7-6C044FE73368</RequestId>
</BindMFADeviceResponse>
```

JSON 格式

```
{
  "RequestId": "04F0F334-1335-436C-A1D7-6C044FE73368"
}
```

错误码

访问[错误中心](#)查看更多错误码。

4.18 UnbindMFADevice

调用UnbindMFADevice接口解绑多因素认证设备。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	UnbindMFADevice	系统规定参数。取值：UnbindMFADevice
UserName	String	是	zhangq****	指定用户名。

返回数据

名称	类型	示例值	描述
MFADevice			多因素认证设备信息。
SerialNumber	String	acs:ram:: 1234567890 12****:mfa/ device002	设备序列号。
RequestId	String	04F0F334-1335 -436C-A1D7- 6C044FE73368	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=UnbindMFADevice  
&UserName=zhangq****  
&<公共请求参数>
```

正常返回示例

XML 格式

```
<UnbindMFADeviceResponse>  
  <RequestId>04F0F334-1335-436C-A1D7-6C044FE73368</RequestId>  
  <MFADevice>  
    <SerialNumber>acs:ram::123456789012****:mfa/device002</  
SerialNumber>  
  </MFADevice>  
</UnbindMFADeviceResponse>
```

JSON 格式

```
{  
  "RequestId": "04F0F334-1335-436C-A1D7-6C044FE73368",  
  "MFADevice": {  
    "SerialNumber": "acs:ram::123456789012****:mfa/device002"  
  }  
}
```

错误码

访问[错误中心](#)查看更多错误码。

4.19 GetUserMFAInfo

调用GetUserMFAInfo接口获取指定RAM用户的多因素认证设备信息。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	GetUserMFAInfo	系统规定参数。取值：GetUserMFAInfo
UserName	String	是	zhangq****	指定用户名。

返回数据

名称	类型	示例值	描述
MFADevice			MFA设备信息。
SerialNumber	String	Virtual-MFA-device002	设备序列号。
RequestId	String	04F0F334-1335-436C-A1D7-6C044FE73368	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=GetUserMFAInfo
&UserName=zhangq****
&<公共请求参数>
```

正常返回示例

XML 格式

```
<GetUserMFAInfoResponse>
  <RequestId>04F0F334-1335-436C-A1D7-6C044FE73368</RequestId>
  <MFADevice>
    <SerialNumber>Virtual-MFA-device002</SerialNumber>
  </MFADevice>
```

```
</GetUserMFAInfoResponse>
```

JSON 格式

```
{
  "RequestId": "04F0F334-1335-436C-A1D7-6C044FE73368",
  "MFADevice": {
    "SerialNumber": "Virtual-MFA-device002"
  }
}
```

错误码

访问[错误中心](#)查看更多错误码。

4.20 ChangePassword

调用ChangePassword接口修改RAM用户密码。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	ChangePassword	系统规定参数。取值：ChangePassword
NewPassword	String	是	aw\$2****	指定密码，密码必须符合密码强度要求。 关于密码强度设置接口，请参考 SetPasswordPolicy 。
OldPassword	String	是	12****	旧密码。

返回数据

名称	类型	示例值	描述
RequestId	String	04F0F334-1335-436C-A1D7-6C044FE73368	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=ChangePassword
&NewPassword=aw$2****
&OldPassword=12****
&<公共请求参数>
```

正常返回示例

XML 格式

```
<ChangePassword>
  <RequestId>04F0F334-1335-436C-A1D7-6C044FE73368</RequestId>
</ChangePassword>
```

JSON 格式

```
{
  "RequestId": "04F0F334-1335-436C-A1D7-6C044FE73368"
}
```

错误码

访问[错误中心](#)查看更多错误码。

5 组管理接口

5.1 CreateGroup

调用CreateGroup接口创建用户组。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	CreateGroup	系统规定参数。取值：CreateGroup
Comments	String	否	开发团队	备注信息，最大长度128个字符。
GroupName	String	否	Dev-Team	用户组名称，最大长度64个字符。 格式： <code>^[a-zA-Z0-9\-_]+\$</code> 。

返回数据

名称	类型	示例值	描述
Group			用户组信息。
Comments	String	开发团队	备注信息。
CreateDate	String	2015-01-23T12:33:18Z	创建时间。
GroupName	String	Dev-Team	用户组名称。
RequestId	String	D3F0679E-9757-95DB-AF2D-04D5188C69C5	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=CreateGroup
&GroupName=Dev-Team
&Comments=开发团队
&<公共请求参数>
```

正常返回示例

XML 格式

```
<CreateGroupResponse>
  <RequestId>D3F0679E-9757-95DB-AF2D-04D5188C69C5</RequestId>
  <Group>
    <GroupName>Dev-Team</GroupName>
    <Comments>开发团队</Comments>
    <CreateDate>2015-01-23T12:33:18Z</CreateDate>
  </Group>
</CreateGroupResponse>
```

JSON 格式

```
{
  "RequestId": "D3F0679E-9757-95DB-AF2D-04D5188C69C5",
  "Group": {
    "Comments": "开发团队",
    "GroupName": "Dev-Team",
    "CreateDate": "2015-01-23T12:33:18Z"
  }
}
```

错误码

访问[错误中心](#)查看更多错误码。

5.2 GetGroup

调用GetGroup接口获取用户组信息。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	GetGroup	系统规定参数。取值：GetGroup

名称	类型	是否必选	示例值	描述
GroupName	String	是	Dev-Team	用户组名称。 格式: <code>^[a-zA-Z0-9_-]+\$</code> 。

返回数据

名称	类型	示例值	描述
Group			用户组信息。
Comments	String	开发团队	备注信息。
CreateDate	String	2015-01-23T12:33:18Z	创建时间。
GroupName	String	Dev-Team	用户组名称。
UpdateDate	String	2015-02-11T03:15:21Z	更新时间。
RequestId	String	D4065824-E422-3ED6-68B1-1AF7D5C7804C	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=GetGroup
&GroupName=Dev-Team
&<公共请求参数>
```

正常返回示例

XML 格式

```
<GetGroupResponse>
  <RequestId>D4065824-E422-3ED6-68B1-1AF7D5C7804C</RequestId>
  <Group>
    <GroupName>Dev-Team</GroupName>
    <Comments>开发团队</Comments>
    <CreateDate>2015-01-23T12:33:18Z</CreateDate>
    <UpdateDate>2015-02-11T03:15:21Z</UpdateDate>
  </Group>
```

```
</GetGroupResponse>
```

JSON 格式

```
{
  "RequestId": "D4065824-E422-3ED6-68B1-1AF7D5C7804C",
  "Group": {
    "Comments": "开发团队",
    "GroupName": "Dev-Team",
    "UpdateDate": "2015-02-11T03:15:21Z",
    "CreateDate": "2015-01-23T12:33:18Z"
  }
}
```

错误码

访问[错误中心](#)查看更多错误码。

5.3 UpdateGroup

调用UpdateGroup接口更新用户组信息。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	UpdateGroup	系统规定参数。取值： UpdateGroup
GroupName	String	是	Dev-Team	指定用户组名称。 格式： <code>^[a-zA-Z0-9_-]+\$</code> 。
NewComments	String	否	开发团队	新的备注信息，最大长度128个字符。
NewGroupName	String	否	NewDev-Team	新的用户组名称。 格式： <code>^[a-zA-Z0-9_-]+\$</code> 。

返回数据

名称	类型	示例值	描述
Group			用户组信息。
Comments	String	开发团队	备注信息。
CreateDate	String	2015-01-23T12:33:18Z	创建时间。
GroupName	String	NewDev-Team	用户组名称。
UpdateDate	String	2015-01-23T12:33:18Z	更新时间。
RequestId	String	EC6647CC-0A36-EC7A-BA72-CC81BF3DE182	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=UpdateGroup
&GroupName=Dev-Team
&NewGroupName=NewDev-Team
&<公共请求参数>
```

正常返回示例

XML 格式

```
<UpdateGroupResponse>
  <RequestId>EC6647CC-0A36-EC7A-BA72-CC81BF3DE182</RequestId>
  <Group>
    <GroupName>NewDev-Team</GroupName>
    <Comments>开发团队</Comments>
    <CreateDate>2015-01-23T12:33:18Z</CreateDate>
    <UpdateDate>2015-01-23T12:33:18Z</UpdateDate>
  </Group>
</UpdateGroupResponse>
```

JSON 格式

```
{
  "RequestId": "EC6647CC-0A36-EC7A-BA72-CC81BF3DE182",
  "Group": {
    "Comments": "开发团队",
    "GroupName": "NewDev-Team",
    "UpdateDate": "2015-01-23T12:33:18Z",
    "CreateDate": "2015-01-23T12:33:18Z"
  }
}
```

}

错误码

访问[错误中心](#)查看更多错误码。

5.4 ListGroups

调用ListGroups接口列出所有的用户组。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	ListGroups	系统规定参数。取值：ListGroups
Marker	String	否	EXAMPLE	当请求的返回结果被截断时，可以使用Marker获取从当前截断位置之后的内容。
MaxItems	Integer	否	100	指定返回结果的条数，当返回结果达到MaxItems限制被截断时，返回参数IsTruncated将等于true。 取值范围：1~1000，默认值：100。

返回数据

名称	类型	示例值	描述
Groups			用户组信息。
Comments	String	开发团队	备注信息。
CreateDate	String	2015-01-23T12:33:18Z	创建时间。
GroupName	String	Dev-Team	用户组名称。

名称	类型	示例值	描述
UpdateDate	String	2015-01-23T12:33:18Z	更新时间。
IsTruncated	Boolean	true	请求返回结果是否被截断。
Marker	String	EXAMPLE	当IsTruncated为true时才有此字段，当返回true时，需要继续调用此接口，并且使用Marker获取截断后的内容。
RequestId	String	065527AA-2F2E-AD7C-7484-F2626CFE4934	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=ListGroups
&<公共请求参数>
```

正常返回示例

XML 格式

```
<ListGroupsResponse>
  <RequestId>065527AA-2F2E-AD7C-7484-F2626CFE4934</RequestId>
  <IsTruncated>true</IsTruncated>
  <Marker>EXAMPLE</Marker>
  <Groups>
    <Group>
      <GroupName>Dev-Team</GroupName>
      <Comments>开发团队</Comments>
      <CreateDate>2015-01-23T12:33:18Z</CreateDate>
      <UpdateDate>2015-01-23T12:33:18Z</UpdateDate>
    </Group>
    <Group>
      <GroupName>QA-Team</GroupName>
      <Comments>测试团队</Comments>
      <CreateDate>2015-02-18T17:22:08Z</CreateDate>
      <UpdateDate>2015-02-18T17:22:08Z</UpdateDate>
    </Group>
  </Groups>
</ListGroupsResponse>
```

JSON 格式

```
{
  "Groups":{
```

```

"Group": [
  {
    "Comments": "开发团队",
    "GroupName": "Dev-Team",
    "UpdateDate": "2015-01-23T12:33:18Z",
    "CreateDate": "2015-01-23T12:33:18Z"
  },
  {
    "Comments": "测试团队",
    "GroupName": "QA-Team",
    "UpdateDate": "2015-02-18T17:22:08Z",
    "CreateDate": "2015-02-18T17:22:08Z"
  }
],
"IsTruncated": true,
"RequestId": "065527AA-2F2E-AD7C-7484-F2626CFE4934",
"Marker": "EXAMPLE"
}

```

错误码

访问[错误中心](#)查看更多错误码。

5.5 DeleteGroup

调用DeleteGroup接口删除指定的用户组。

- 删除用户组前，需要确保用户组没有绑定任何权限策略。
- 删除用户组前，需要确保用户组内没有用户。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	DeleteGroup	系统规定参数。取值：DeleteGroup
GroupName	String	是	Dev-Team	指定用户组名称。 格式： <code>^[a-zA-Z0-9\-_]+\$</code> 。

返回数据

名称	类型	示例值	描述
RequestId	String	FCF40AB5-881C-A0F9-334C-B0AD423AA69D	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=DeleteGroup
&GroupName=Dev-Team
&<公共请求参数>
```

正常返回示例

XML 格式

```
<DeleteGroupResponse>
  <RequestId>FCF40AB5-881C-A0F9-334C-B0AD423AA69D</RequestId>
</DeleteGroupResponse>
```

JSON 格式

```
{
  "RequestId": "FCF40AB5-881C-A0F9-334C-B0AD423AA69D"
}
```

错误码

访问[错误中心](#)查看更多错误码。

5.6 AddUserToGroup

调用AddUserToGroup接口将RAM用户添加到指定的用户组。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	AddUserToGroup	系统规定参数。取值：AddUserToGroup

名称	类型	是否必选	示例值	描述
GroupName	String	是	Dev-Team	用户组名称。
UserName	String	是	zhangq****	用户名称。

返回数据

名称	类型	示例值	描述
RequestId	String	1B968853-B423-63A6-FE1F-45E81BC2AD61	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=AddUserToGroup
&UserName=zhangq****
&GroupName=Dev-Team
&<公共请求参数>
```

正常返回示例

XML 格式

```
<AddUserToGroupResponse>
  <RequestId>1B968853-B423-63A6-FE1F-45E81BC2AD61</RequestId>
</AddUserToGroupResponse>
```

JSON 格式

```
{
  "RequestId": "1B968853-B423-63A6-FE1F-45E81BC2AD61"
}
```

错误码

访问[错误中心](#)查看更多错误码。

5.7 RemoveUserFromGroup

调用RemoveUserFromGroup接口将RAM用户从用户组中移除。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	RemoveUserFromGroup	系统规定参数。取值：RemoveUserFromGroup
GroupName	String	是	Dev-Team	用户组名称。
UserName	String	是	zhangq****	RAM用户名称。

返回数据

名称	类型	示例值	描述
RequestId	String	A07EF215-B9B3-8CB2-2899-3F9575C6E320	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=RemoveUserFromGroup
&UserName=zhangq****
&GroupName=Dev-Team
&<公共请求参数>
```

正常返回示例

XML 格式

```
<RemoveUserFromGroupResponse>
  <RequestId>A07EF215-B9B3-8CB2-2899-3F9575C6E320</RequestId>
</RemoveUserFromGroupResponse>
```

JSON 格式

```
{
```

```
"RequestId": "A07EF215-B9B3-8CB2-2899-3F9575C6E320"
}
```

错误码

访问[错误中心](#)查看更多错误码。

5.8 ListGroupsForUser

调用ListGroupsForUser接口列出指定RAM用户所加入的用户组信息。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	ListGroupsForUser	系统规定参数。取值：ListGroupsForUser
UserName	String	是	zhangq****	用户名称。

返回数据

名称	类型	示例值	描述
Groups			用户组信息。
Comments	String	开发团队	备注信息。
GroupName	String	Dev-Team	用户组名称。
JoinDate	String	2015-01-23T12:33:18Z	加入时间。
RequestId	String	0945BF6F-54A4-C384-C89C-C500DCEE367C	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=ListGroupsForUser
```



```
&UserName=zhangq****  
&<公共请求参数>
```

正常返回示例

XML 格式

```
<ListGroupForUserResponse>  
  <RequestId>0945BF6F-54A4-C384-C89C-C500DCEE367C</RequestId>  
  <Groups>  
    <Group>  
      <GroupName>Dev-Team</GroupName>  
      <Comments>开发团队</Comments>  
      <JoinDate>2015-01-23T12:33:18Z</JoinDate>  
    </Group>  
    <Group>  
      <GroupName>QA-Team</GroupName>  
      <Comments>测试团队</Comments>  
      <JoinDate>2015-02-18T17:22:08Z</JoinDate>  
    </Group>  
  </Groups>  
</ListGroupForUserResponse>
```

JSON 格式

```
{  
  "Groups":{  
    "Group":[  
      {  
        "Comments":"开发团队",  
        "GroupName":"Dev-Team",  
        "JoinDate":"2015-01-23T12:33:18Z"  
      },  
      {  
        "Comments":"测试团队",  
        "GroupName":"QA-Team",  
        "JoinDate":"2015-02-18T17:22:08Z"  
      }  
    ]  
  },  
  "RequestId":"0945BF6F-54A4-C384-C89C-C500DCEE367C"  
}
```

错误码

访问[错误中心](#)查看更多错误码。

5.9 ListUsersForGroup

调用ListUsersForGroup接口列出指定用户组所包含的RAM用户。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	ListUsersForGroup	系统规定参数。取值：ListUsersForGroup
GroupName	String	是	Dev-Team	用户组名称。
Marker	String	否	EXAMPLE	当请求的返回结果被截断时，可以使用Marker获取从当前截断位置之后的内容。
MaxItems	Integer	否	100	指定返回结果的条数。当返回结果达到MaxItems限制被截断时，返回参数IsTruncated将等于true。 取值范围：1 ~ 1000，默认值：100。

返回数据

名称	类型	示例值	描述
IsTruncated	Boolean	true	请求返回结果是否被截断。
Marker	String	EXAMPLE	当IsTruncated为true时才有此字段。当返回true时，需要继续调用此接口，并且使用Marker获取截断后的内容。
RequestId	String	4B450CA1-36E8-4AA2-8461-86B42BF4CC4E	请求ID。
Users			用户信息。
DisplayName	String	张强	显示名称。
JoinDate	String	2015-01-23T12:33:18Z	加入日期。

名称	类型	示例值	描述
UserName	String	zhangqiang	用户名称。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=ListUsersForGroup
&GroupName=dev
&<公共请求参数>
```

正常返回示例

XML 格式

```
<ListUsersForGroupResponse>
  <RequestId>5756784B-79C4-E82E-24C2-FC3E171E5AB3</RequestId>
  <Users>
    <User>
      <UserName>zhangqiang</UserName>
      <DisplayName>张强</DisplayName>
      <JoinDate>2015-01-23T12:33:18Z</JoinDate>
    </User>
    <User>
      <UserName>lili</UserName>
      <DisplayName>李丽</DisplayName>
      <JoinDate>2015-02-18T17:22:08Z</JoinDate>
    </User>
  </Users>
</ListUsersForGroupResponse>
```

JSON 格式

```
{
  "Users": {
    "User": [
      {
        "JoinDate": "2015-01-23T12:33:18Z",
        "UserName": "zhangqiang",
        "DisplayName": "张强"
      },
      {
        "JoinDate": "2015-02-18T17:22:08Z",
        "UserName": "lili",
        "DisplayName": "李丽"
      }
    ]
  },
  "RequestId": "4B450CA1-36E8-4AA2-8461-86B42BF4CC4E"
}
```

错误码

访问[错误中心](#)查看更多错误码。

6 角色管理接口


6.1 CreateRole

调用CreateRole接口创建角色。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	CreateRole	系统规定参数。取值：CreateRole
RoleName	String	是	ECSAdmin	指定角色名，最多包含64个字符。 格式： <code>^[a-zA-Z0-9.@-]+\$</code> 。
AssumeRolePolicyDocument	String	是	<code>{"Statement": [{"Action": "sts:AssumeRole", "Effect": "Allow", "Principal": {"RAM": "acs:ram::123456789012****:root"}}, {"Version": "1"}]}</code>	一个策略文本。指定允许扮演该角色的一个或多个主体，这个主体可以是阿里云账号、阿里云服务或身份提供商。  说明： RAM用户不能扮演可信实体为阿里云服务的RAM角色。
Description	String	否	ECS管理角色	角色描述，最大长度为1024个字符。

AssumeRolePolicyDocument样例说明

- 以下策略表示：允许扮演该RAM角色的可信实体为云账号(AccountID=123456789012****)下被授权的任何RAM用户。

```
{
  "Statement": [
    {
      "Action": "sts:AssumeRole",
```

```

    "Effect": "Allow",
    "Principal": {
      "RAM": [
        "acs:ram::123456789012****:root"
      ]
    }
  ],
  "Version": "1"
}

```

- 以下策略表示：允许扮演该RAM角色的可信实体为云账号(AccountID=123456789012****)下被授权的RAM用户 (testuser)。



说明:

创建此角色时，请确保已创建好RAM用户testuser(其UPN为：testuser@123456789012****.onaliyun.com)。

```

{
  "Statement": [
    {
      "Action": "sts:AssumeRole",
      "Effect": "Allow",
      "Principal": {
        "RAM": [
          "acs:ram::123456789012****:user/testuser"
        ]
      }
    }
  ],
  "Version": "1"
}

```

- 以下策略表示：允许扮演该RAM角色的可信实体为当前云账号下的ECS服务。

```

{
  "Statement": [
    {
      "Action": "sts:AssumeRole",
      "Effect": "Allow",
      "Principal": {
        "Service": [
          "ecs.aliyuncs.com"
        ]
      }
    }
  ],
  "Version": "1"
}

```

- 以下策略表示：允许扮演该RAM角色的可信实体为当前云账号(AccountID=123456789012****)中的身份提供商 (testprovider) 下的用户。

```

{

```

```

    "Statement": [
      {
        "Action": "sts:AssumeRole",
        "Effect": "Allow",
        "Principal": {
          "Federated": [
            "acs:ram::123456789012****:saml-provider/
testprovider"
          ]
        },
        "Condition": {
          "StringEquals": {
            "saml:recipient": "https://signin.aliyun.com/saml
-role/sso"
          }
        }
      }
    ],
    "Version": "1"
  }
}

```

返回数据

名称	类型	示例值	描述
RequestId	String	04F0F334-1335-436C-A1D7-6C044FE73368	请求ID。
Role			角色信息。
Arn	String	acs:ram::123456789012****:role/ECSAdmin	角色的资源描述符。
AssumeRolePolicyDocument	String	{ "Statement": [{ "Action": "sts:AssumeRole", "Effect": "Allow", "Principal": { "RAM": "acs:ram::123456789012****:root" } }], "Version": "1" }	扮演角色的权限策略。
CreateDate	String	2015-01-23T12:33:18Z	创建时间。
Description	String	ECS管理角色	角色的文字描述。

名称	类型	示例值	描述
RoleId	String	901234567890 ****	角色ID。
RoleName	String	ECSAdmin	角色名称。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=CreateRole
&RoleName=ECSAdmin
&AssumeRolePolicyDocument={"Statement":[{"Action":"sts:AssumeRole",
Effect:"Allow","Principal":{"RAM":{"acs:ram::123456789012****:root
"}}], "Version":"1"}
&Description=ECS管理角色
&<公共请求参数>
```

正常返回示例

XML 格式

```
<CreateRoleResponse>
  <RequestId>04F0F334-1335-436C-A1D7-6C044FE73368</RequestId>
  <Role>
    <RoleId>901234567890****</RoleId>
    <RoleName>ECSAdmin</RoleName>
    <Arn>acs:ram::123456789012****:role/ECSAdmin</Arn>
    <Description>ECS管理角色</Description>
    <AssumeRolePolicyDocument>{ "Statement": [ { "Action": "
sts:AssumeRole", "Effect": "Allow", "Principal": { "RAM": "acs:ram::
123456789012****:root" } } ], "Version": "1" }</AssumeRolePolicyDocu
ment>
    <CreateDate>2015-01-23T12:33:18Z</CreateDate>
  </Role>
</CreateRoleResponse>
```

JSON 格式

```
{
  "RequestId": "04F0F334-1335-436C-A1D7-6C044FE73368",
  "Role": {
    "RoleName": "ECSAdmin",
    "Description": "ECS管理角色",
    "AssumeRolePolicyDocument": "{ \"Statement\": [ { \"Action\": \"sts:
AssumeRole\", \"Effect\": \"Allow\", \"Principal\": { \"RAM\": \"acs:
ram::123456789012****:root\" } } ], \"Version\": \"1\" }",
    "Arn": "acs:ram::123456789012****:role/ECSAdmin",
    "CreateDate": "2015-01-23T12:33:18Z",
    "RoleId": "901234567890****"
  }
}
```

```
}

```

错误码

访问[错误中心](#)查看更多错误码。

6.2 GetRole

调用GetRole接口获取角色信息。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	GetRole	系统规定参数。取值：GetRole
RoleName	String	是	ECSAdmin	指定角色名，最多包含64个字符。 格式： <code>^[a-zA-Z0-9\.\@\-]+\.</code>

返回数据

名称	类型	示例值	描述
RequestId	String	04F0F334-1335-436C-A1D7-6C044FE73368	请求ID。
Role			角色信息。
Arn	String	acs:ram::123456789012****:role/ECSAdmin	角色的资源描述符。

名称	类型	示例值	描述
AssumeRolePolicyDocument	String	{ "Statement": [{ "Action": "sts:AssumeRole", "Effect": "Allow", "Principal": { "RAM": "acs:ram::123456789012****:root" } }], "Version": "1" }	扮演角色的权限策略。
CreateDate	String	2015-01-23T12:33:18Z	创建时间。
Description	String	ECS管理角色	角色的文字描述。
RoleId	String	901234567890****	角色ID。
RoleName	String	ECSAdmin	角色名称。
UpdateDate	String	2015-01-23T12:33:18Z	更新时间。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=GetRole
&RoleName=ECSAdmin
&<公共请求参数>
```

正常返回示例

XML 格式

```
<GetRoleResponse>
  <RequestId>04F0F334-1335-436C-A1D7-6C044FE73368</RequestId>
  <Role>
    <RoleId>901234567890****</RoleId>
    <RoleName>ECSAdmin</RoleName>
    <Arn>acs:ram::123456789012****:role/ECSAdmin</Arn>
    <Description>ECS管理角色</Description>
    <AssumeRolePolicyDocument>{ "Statement": [ { "Action": "
sts:AssumeRole", "Effect": "Allow", "Principal": { "RAM": "acs:ram::
123456789012****:root" } } ], "Version": "1" }</AssumeRolePolicyDocu
ment>
    <CreateDate>2015-01-23T12:33:18Z</CreateDate>
    <UpdateDate>2015-01-23T12:33:18Z</UpdateDate>
  </Role>
```

```
</GetRoleResponse>
```

JSON 格式

```
{
  "RequestId": "04F0F334-1335-436C-A1D7-6C044FE73368",
  "Role": {
    "RoleName": "ECSAdmin",
    "Description": "ECS管理角色",
    "AssumeRolePolicyDocument": "{ \"Statement\": [ { \"Action\": \"sts:AssumeRole\", \"Effect\": \"Allow\", \"Principal\": { \"RAM\": \"acs:ram::123456789012****:root\" } } ], \"Version\": \"1\" }",
    "UpdateDate": "2015-01-23T12:33:18Z",
    "Arn": "acs:ram::123456789012****:role/ECSAdmin",
    "CreateDate": "2015-01-23T12:33:18Z",
    "RoleId": "901234567890****"
  }
}
```

错误码

访问[错误中心](#)查看更多错误码。

6.3 UpdateRole

调用UpdateRole接口更新角色信息。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	UpdateRole	系统规定参数。取值：UpdateRole
NewAssumeRolePolicyDocument	String	否	{ "Statement": [{ "Action": "sts:AssumeRole", "Effect": "Allow", "Principal": { "RAM": "acs:ram::123456789012345678:root" } }], "Version": "1" }	指定可以扮演此角色的身份。

名称	类型	是否必选	示例值	描述
RoleName	String	否	ECSAdmin	指定角色名，最多包含64个字符。 格式： <code>^[a-zA-Z0-9\.\@\-]+\.</code>

返回数据

名称	类型	示例值	描述
RequestId	String	04F0F334-1335-436C-A1D7-6C044FE73368	请求ID。
Role			角色信息。
Arn	String	acs:ram::123456789012****:role/ECSAdmin	角色的资源描述符。
AssumeRolePolicyDocument	String	{ "Statement": [{ "Action": "sts:AssumeRole", "Effect": "Allow", "Principal": { "RAM": "acs:ram::123456789012****:root" } }], "Version": "1" }	扮演角色的权限策略。
CreateDate	String	2015-01-23T12:33:18Z	创建时间。
Description	String	ECS管理角色	角色的文字描述。
RoleId	String	901234567890****	角色ID。
RoleName	String	ECSAdmin	角色名称。
UpdateDate	String	2015-01-23T12:33:18Z	更新时间。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=UpdateRole
&RoleName=ECSAdmin
&NewAssumeRolePolicyDocument={ "Statement": [ { "Action": "sts:
AssumeRole", "Effect": "Allow", "Principal": { "RAM": "acs:ram::
123456789012****:root" } } ], "Version": "1" }
&<公共请求参数>
```

正常返回示例

XML 格式

```
<UpdateRoleResponse>
  <RequestId>04F0F334-1335-436C-A1D7-6C044FE73368</RequestId>
  <Role>
    <RoleId>901234567890****</RoleId>
    <RoleName>ECSAdmin</RoleName>
    <Arn>acs:ram::123456789012****:role/ECSAdmin</Arn>
    <Description>ECS管理角色</Description>
    <AssumeRolePolicyDocument>{ "Statement": [ { "Action": "
sts:AssumeRole", "Effect": "Allow", "Principal": { "RAM": "acs:ram::
123456789012****:root" } } ], "Version": "1" }</AssumeRolePolicyDocu
ment>
    <CreateDate>2015-01-23T12:33:18Z</CreateDate>
    <UpdateDate>2015-01-23T12:33:18Z</UpdateDate>
  </Role>
</UpdateRoleResponse>
```

JSON 格式

```
{
  "RequestId": "04F0F334-1335-436C-A1D7-6C044FE73368",
  "Role": {
    "RoleName": "ECSAdmin",
    "Description": "ECS管理角色",
    "AssumeRolePolicyDocument": { "Statement": [ { "Action": "sts:
AssumeRole", "Effect": "Allow", "Principal": { "RAM": "acs:
ram::123456789012****:root" } } ], "Version": "1" },
    "UpdateDate": "2015-01-23T12:33:18Z",
    "Arn": "acs:ram::123456789012****:role/ECSAdmin",
    "CreateDate": "2015-01-23T12:33:18Z",
    "RoleId": "901234567890****"
  }
}
```

错误码

访问[错误中心](#)查看更多错误码。

6.4 ListRoles

调用ListRoles接口列出角色。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	ListRoles	系统规定参数。取值：ListRoles
Marker	String	否	EXAMPLE	当请求的返回结果被截断时，可以使用Marker获取从当前截断位置之后的内容。
MaxItems	Integer	否	100	指定返回结果的条数。当返回结果达到MaxItems限制被截断时，返回参数IsTruncated将等于true。 取值范围：1 ~ 1000，默认值：100。

返回数据

名称	类型	示例值	描述
IsTruncated	Boolean	true	请求返回结果是否被截断。
Marker	String	EXAMPLE	当IsTruncated为true时才有此字段。当返回true时，需要继续调用此接口，并且使用Marker获取截断后的内容。
RequestId	String	7B8A4E7D-6CFF-471D-84DF-195A7A241ECB	请求ID。
Roles			角色信息。

名称	类型	示例值	描述
Arn	String	acs:ram:: 1234567890 12****:role/ ECSAdmin	角色的资源描述符。
CreateDate	String	2015-01-23T12:33 :18Z	创建时间。
Description	String	ECS管理角色	角色的备注。
RoleId	String	901234567890 ****	角色ID。
RoleName	String	ECSAdmin	角色名称。
UpdateDate	String	2015-01-23T12:33 :18Z	更新时间。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=ListRoles
&<公共请求参数>
```

正常返回示例

XML 格式

```
<ListRolesResponse>
  <RequestId>7B8A4E7D-6CFF-471D-84DF-195A7A241ECB</RequestId>
  <IsTruncated>>true</IsTruncated>
  <Marker>EXAMPLE</Marker>
  <Roles>
    <Role>
      <RoleId>901234567890****</RoleId>
      <RoleName>ECSAdmin</RoleName>
      <Arn>acs:ram::123456789012****:role/ECSAdmin</Arn>
      <Description>ECS管理角色</Description>
      <CreateDate>2015-01-23T12:33:18Z</CreateDate>
      <UpdateDate>2015-01-23T12:33:18Z</UpdateDate>
    </Role>
    <Role>
      <RoleId>901234567890****</RoleId>
      <RoleName>OSSReadOnlyAccess</RoleName>
      <Arn>acs:ram::123456789012****:role/OSSReadOnlyAccess</Arn>
      <Description>OSS只读访问角色</Description>
      <CreateDate>2015-01-23T12:33:18Z</CreateDate>
      <UpdateDate>2015-01-23T12:33:18Z</UpdateDate>
    </Role>
  </Roles>
```

```
</ListRolesResponse>
```

JSON 格式

```
{
  "IsTruncated":true,
  "RequestId":"7B8A4E7D-6CFF-471D-84DF-195A7A241ECB",
  "Roles":{
    "Role":[
      {
        "RoleName":"ECSAdmin",
        "Description":"ECS管理角色",
        "UpdateDate":"2015-01-23T12:33:18Z",
        "Arn":"acs:ram::123456789012****:role/ECSAdmin",
        "CreateDate":"2015-01-23T12:33:18Z",
        "RoleId":"901234567890****"
      },
      {
        "RoleName":"OSSReadOnlyAccess",
        "Description":"OSS只读访问角色",
        "UpdateDate":"2015-01-23T12:33:18Z",
        "Arn":"acs:ram::123456789012****:role/OSSReadOnlyAccess",
        "CreateDate":"2015-01-23T12:33:18Z",
        "RoleId":"901234567890****"
      }
    ]
  },
  "Marker":"EXAMPLE"
}
```

错误码

访问[错误中心](#)查看更多错误码。

6.5 DeleteRole

调用DeleteRole接口删除指定的角色。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	DeleteRole	系统规定参数。取值：DeleteRole
RoleName	String	是	ECSAdmin	指定角色名称。

返回数据

名称	类型	示例值	描述
RequestId	String	898FAB24-7509-43EE-A287-086FE4C44394	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=DeleteRole
&RoleName=ECSAdmin
&<公共请求参数>
```

正常返回示例

XML 格式

```
<DeleteRoleResponse>
  <RequestId>898FAB24-7509-43EE-A287-086FE4C44394</RequestId>
</DeleteRoleResponse>
```

JSON 格式

```
{
  "RequestId": "898FAB24-7509-43EE-A287-086FE4C44394"
}
```

错误码

访问[错误中心](#)查看更多错误码。

7 权限策略管理接口

7.1 CreatePolicy

调用CreatePolicy接口创建一个权限策略。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	CreatePolicy	系统规定参数。取值：CreatePolicy
Description	String	否	OSS管理员权限	描述，最大长度1024个字符。
PolicyDocument	String	否	{ "Statement": [{"Action": "oss:*", "Effect": "Allow", "Resource": ["acs:oss:*:*:*"], }], "Version": "1"} }	权限策略内容，最大长度2048字节。
PolicyName	String	否	OSS-Administrator	权限策略名称，最多包含128个字符。 格式： <code>^[a-zA-Z0-9\-_]+\$</code> 。

返回数据

名称	类型	示例值	描述
Policy			权限策略信息。
CreateDate	String	2015-01-23T12:33:18Z	创建时间。

名称	类型	示例值	描述
DefaultVersion	String	v1	默认版本。
Description	String	OSS管理员权限	描述。
PolicyName	String	OSS-Administrator	权限策略名称。
PolicyType	String	Custom	权限策略类型。
RequestId	String	9B34724D-54B0-4A51-B34D-4512372FE1BE	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=CreatePolicy
&PolicyName=OSS-Administrator
&PolicyDocument={"Statement":[{"Action":["oss:*"],"Effect":"Allow","Resource":["acs:oss:*:*:*"]}], "Version":"1"}
&Description=OSS管理员权限
&<公共请求参数>
```

正常返回示例

XML 格式

```
<CreatePolicyResponse>
  <RequestId>9B34724D-54B0-4A51-B34D-4512372FE1BE</RequestId>
  <Policy>
    <PolicyName>OSS-Administrator</PolicyName>
    <PolicyType>Custom</PolicyType>
    <Description>OSS管理员权限</Description>
    <DefaultVersion>v1</DefaultVersion>
    <CreateDate>2015-01-23T12:33:18Z</CreateDate>
  </Policy>
</CreatePolicyResponse>
```

JSON 格式

```
{
  "Policy":{
    "Description":"OSS管理员权限",
    "PolicyName":"OSS-Administrator",
    "CreateDate":"2015-01-23T12:33:18Z",
    "DefaultVersion":"v1",
    "PolicyType":"Custom"
  },
  "RequestId":"9B34724D-54B0-4A51-B34D-4512372FE1BE"
```

```
}

```

错误码

访问[错误中心](#)查看更多错误码。

7.2 GetPolicy

调用GetPolicy接口获取指定的权限策略信息。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	GetPolicy	系统规定参数。取值：GetPolicy
PolicyName	String	是	OSS-Administrator	指定权限策略名称。
PolicyType	String	是	Custom	指定Policy的类型，取值System或Custom。

返回数据

名称	类型	示例值	描述
DefaultPolicyVersion			权限策略的默认版本。
CreateDate	String	2015-01-23T12:33:18Z	创建时间。
IsDefaultVersion	Boolean	true	DefaultPolicyVersion返回的数据结构固定为默认版本，因此isDefault的取值固定为ture。
PolicyDocument	String	{ "Statement": [{ "Action": ["oss:*"], "Effect": "Allow", "Resource": ["acs:oss:*:*:*"] }], "Version": "1" }	权限策略内容。

名称	类型	示例值	描述
VersionId	String	v1	权限策略标识。
Policy			权限策略的基本信息。
Attachment Count	Integer	0	引用次数。
CreateDate	String	2015-01-23T12:33:18Z	创建时间。
DefaultVersion	String	v1	默认版本。
Description	String	OSS管理员权限	权限策略描述。
PolicyDocument	String	N/A	已废弃。
PolicyName	String	OSS-Administrator	权限策略名称。
PolicyType	String	Custom	权限策略类型。
UpdateDate	String	2015-01-23T12:33:18Z	修改时间。
RequestId	String	697852FB-50D7-44D9-9774-530C31EAC572	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=GetPolicy
&PolicyName=OSS-Administrator
&PolicyType=Custom
&<公共请求参数>
```

正常返回示例

XML 格式

```
<GetPolicyResponse>
  <RequestId>697852FB-50D7-44D9-9774-530C31EAC572</RequestId>
  <Policy>
    <PolicyName>OSS-Administrator</PolicyName>
    <PolicyType>Custom</PolicyType>
    <Description>OSS管理员权限</Description>
```

```
<DefaultVersion>v1</DefaultVersion>
<CreateDate>2015-01-23T12:33:18Z</CreateDate>
<UpdateDate>2015-01-23T12:33:18Z</UpdateDate>
<AttachmentCount>0</AttachmentCount>
</Policy>
<DefaultPolicyVersion>
<VersionId>v1</VersionId>
<IsDefaultVersion>true</IsDefaultVersion>
<CreateDate>2015-01-23T12:33:18Z</CreateDate>
<PolicyDocument>{ "Statement": [{ "Action": ["oss:*"], "Effect":
"Allow", "Resource": ["acs:oss:*:*:*"]}], "Version": "1"}</PolicyDocu
ment>
</DefaultPolicyVersion>
</GetPolicyResponse>
```

JSON 格式

```
{
  "Policy":{
    "AttachmentCount":0,
    "Description":"OSS管理员权限",
    "PolicyName":"OSS-Administrator",
    "UpdateDate":"2015-01-23T12:33:18Z",
    "CreateDate":"2015-01-23T12:33:18Z",
    "DefaultVersion":"v1",
    "PolicyType":"Custom"
  },
  "DefaultPolicyVersion":{
    "IsDefaultVersion":true,
    "VersionId":"v1",
    "PolicyDocument":"{ \"Statement\": [{ \"Action\": [\"oss:*\"], \"
Effect\": \"Allow\", \"Resource\": [\"acs:oss:*:*:*\"]}], \"Version\":
\"1\"}",
    "CreateDate":"2015-01-23T12:33:18Z"
  },
  "RequestId":"697852FB-50D7-44D9-9774-530C31EAC572"
}
```

错误码

访问[错误中心](#)查看更多错误码。

7.3 DeletePolicy

调用DeletePolicy接口删除指定的权限策略。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	DeletePolicy	系统规定参数。取值: DeletePolicy
PolicyName	String	是	OSS-Administrator	指定权限策略名称。

返回数据

名称	类型	示例值	描述
RequestId	String	898FAB24-7509-43EE-A287-086FE4C44394	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=DeletePolicy
&PolicyName=OSS-Administrator
&<公共请求参数>
```

正常返回示例

XML 格式

```
<DeletePolicyResponse>
  <RequestId>898FAB24-7509-43EE-A287-086FE4C44394</RequestId>
</DeletePolicyResponse>
```

JSON 格式

```
{
  "RequestId": "898FAB24-7509-43EE-A287-086FE4C44394"
}
```

错误码

访问[错误中心](#)查看更多错误码。

7.4 ListPolicies

调用ListPolicies接口列出权限策略。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	ListPolicies	系统规定参数。取值：ListPolicies
Marker	String	否	EXAMPLE	当请求的返回结果被截断时，可以使用Marker获取从当前截断位置之后的内容。
MaxItems	Integer	否	100	指定返回结果的条数，当返回结果达到MaxItems限制被截断时，返回参数IsTruncated将等于true。 取值范围：1 ~ 1000，默认值：100。
PolicyType	String	否	System	指定Policy的类型，取值System或Custom，如果没有指定则列出所有权限策略。

返回数据

名称	类型	示例值	描述
IsTruncated	Boolean	true	请求返回结果是否被截断。
Marker	String	EXAMPLE	当IsTruncated为true时才有此字段，当返回true时，需要继续调用此接口，并且使用Marker获取截断后的内容。

名称	类型	示例值	描述
Policies			权限策略信息。
Attachment Count	Integer	3	引用次数。
CreateDate	String	2015-01-23T12:33:18Z	创建时间。
DefaultVersion	String	v1	默认版本。
Description	String	OSS管理员权限	权限策略描述。
PolicyName	String	OSS-Administrator	权限策略名称。
PolicyType	String	System	权限策略类型。
UpdateDate	String	2015-01-23T12:33:18Z	修改时间。
RequestId	String	7B8A4E7D-6CFF-471D-84DF-195A7A241ECB	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=ListPolicies
&<公共请求参数>
```

正常返回示例

XML 格式

```
<ListPoliciesResponse>
  <RequestId>7B8A4E7D-6CFF-471D-84DF-195A7A241ECB</RequestId>
  <IsTruncated>>true</IsTruncated>
  <Marker>EXAMPLE</Marker>
  <Policies>
    <Policy>
      <PolicyName>OSS-Administrator</PolicyName>
      <PolicyType>Custom</PolicyType>
      <Description>OSS管理员权限</Description>
      <DefaultVersion>v1</DefaultVersion>
      <CreateDate>2015-01-23T12:33:18Z</CreateDate>
      <UpdateDate>2015-01-23T12:33:18Z</UpdateDate>
      <AttachmentCount>0</AttachmentCount>
    </Policy>
  </Policies>
</ListPoliciesResponse>
```



```
<Policy>
  <PolicyName>ReadOnlyAccess</PolicyName>
  <PolicyType>System</PolicyType>
  <Description>只读权限</Description>
  <DefaultVersion>v1</DefaultVersion>
  <CreateDate>2015-02-11T18:39:12Z</CreateDate>
  <UpdateDate>2015-02-19T09:43:16Z</UpdateDate>
  <AttachmentCount>0</AttachmentCount>
</Policy>
</Policies>
</ListPoliciesResponse>
```

JSON 格式

```
{
  "Policies":{
    "Policy":[
      {
        "AttachmentCount":3,
        "Description":"OSS管理员权限",
        "PolicyName":"OSS-Administrator",
        "UpdateDate":"2015-01-23T12:33:18Z",
        "CreateDate":"2015-01-23T12:33:18Z",
        "DefaultVersion":"v1",
        "PolicyType":"Custom"
      },
      {
        "AttachmentCount":1,
        "Description":"只读权限",
        "PolicyName":"ReadOnlyAccess",
        "UpdateDate":"2015-02-19T09:43:16Z",
        "CreateDate":"2015-02-11T18:39:12Z",
        "DefaultVersion":"v1",
        "PolicyType":"System"
      }
    ]
  },
  "IsTruncated":true,
  "RequestId":"7B8A4E7D-6CFF-471D-84DF-195A7A241ECB",
  "Marker":"EXAMPLE"
}
```

错误码

访问[错误中心](#)查看更多错误码。

7.5 CreatePolicyVersion

调用CreatePolicyVersion接口为权限策略创建新的版本。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	CreatePolicyVersion	系统规定参数。取值：CreatePolicyVersion
PolicyDocument	String	是	{ "Statement": [{"Action": ["oss:*"], "Effect": "Allow", "Resource": ["acs:oss:*:*:*"], "Version": "1"}]}	权限策略内容，最大长度2048字节。
PolicyName	String	是	OSS-Administrator	权限策略名称。
RotateStrategy	String	否	None	<p>权限策略版本自动化轮转机制，可以删除历史权限策略版本。</p> <p>目前包含：</p> <ul style="list-style-type: none"> · None: 关闭轮转机制。 · DeleteOldestNonDefaultVersionWhenLimitExceeded: 当权限策略版本数量超限时，删除最早且非活跃的版本。 <p>默认值：None</p>
SetAsDefault	Boolean	否	false	是否设置为默认权限策略，默认值为false。

返回数据

名称	类型	示例值	描述
PolicyVersion			新建的权限策略版本的信息。
CreateDate	String	2015-01-23T12:33:18Z	创建时间。

名称	类型	示例值	描述
IsDefaultVersion	Boolean	false	是否默认版本。
PolicyDocument	String	{ "Statement": [{ "Action": ["oss:*"], "Effect": "Allow", "Resource": ["acs:oss:*:*:*"], "Version": "1" }	权限策略内容。
VersionId	String	v3	权限策略标识。
RequestId	String	9B34724D-54B0-4A51-B34D-4512372FE1BE	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=CreatePolicyVersion
&PolicyName=OSS-Administrator
&PolicyDocument={"Statement":[{"Action":["oss:*"],"Effect":"Allow",
Resource":["acs:oss:*:*:*"]}], "Version":"1"}
&<公共请求参数>
```

正常返回示例

XML 格式

```
<CreatePolicyVersionResponse>
  <RequestId>9B34724D-54B0-4A51-B34D-4512372FE1BE</RequestId>
  <PolicyVersion>
    <VersionId>v3</VersionId>
    <IsDefaultVersion>>false</IsDefaultVersion>
    <CreateDate>2015-01-23T12:33:18Z</CreateDate>
    <PolicyDocument>{ "Statement": [ { "Action": ["oss:*"], "
Effect": "Allow", "Resource": ["acs:oss:*:*:*"]}], "Version": "1"}</
PolicyDocument>
  </PolicyVersion>
</CreatePolicyVersionResponse>
```

JSON 格式

```
{
  "RequestId": "9B34724D-54B0-4A51-B34D-4512372FE1BE",
  "PolicyVersion": {
    "IsDefaultVersion": false,
    "VersionId": "v3",
```

```
"PolicyDocument": "{ \"Statement\": [ { \"Action\": [ \"oss:*\" ], \"Effect\": \"Allow\", \"Resource\": [ \"acs:oss:*:*:*\" ] } ], \"Version\": \"1\" }\",
  \"CreateDate\": \"2015-01-23T12:33:18Z\"
}
}
```

错误码

访问[错误中心](#)查看更多错误码。

7.6 GetPolicyVersion

调用GetPolicyVersion接口获取某个权限策略的版本。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	GetPolicyVersion	系统规定参数。取值：GetPolicyVersion
PolicyName	String	是	OSS-Administrator	权限策略名称。
PolicyType	String	是	Custom	指定权限策略的类型，取值System或Custom。
VersionId	String	是	v3	指定目标版本的ID。

返回数据

名称	类型	示例值	描述
PolicyVersion			权限策略版本信息。
CreateDate	String	2015-01-23T12:33:18Z	创建时间。
IsDefaultVersion	Boolean	false	是否默认版本。

名称	类型	示例值	描述
PolicyDocument	String	{ "Statement": [{ "Action": ["oss:*"], "Effect": "Allow", "Resource": ["acs:oss:*:*:*"], "Version": "1" }	权限策略内容。
VersionId	String	v3	权限策略标识。
RequestId	String	9B34724D-54B0-4A51-B34D-4512372FE1BE	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=GetPolicyVersion
&PolicyName=OSS-Administrator
&PolicyType=Custom
&VersionId=v3
&<公共请求参数>
```

正常返回示例

XML 格式

```
<GetPolicyVersionResponse>
  <RequestId>9B34724D-54B0-4A51-B34D-4512372FE1BE</RequestId>
  <PolicyVersion>
    <VersionId>v3</VersionId>
    <IsDefaultVersion>>false</IsDefaultVersion>
    <CreateDate>2015-01-23T12:33:18Z</CreateDate>
    <PolicyDocument>
      { "Statement": [ { "Action": ["oss:*"], "Effect": "Allow",
        "Resource": ["acs:oss:*:*:*"], "Version": "1" }
      }
    </PolicyDocument>
  </PolicyVersion>
</GetPolicyVersionResponse>
```

JSON 格式

```
{
  "RequestId": "9B34724D-54B0-4A51-B34D-4512372FE1BE",
  "PolicyVersion": {
    "IsDefaultVersion": false,
    "VersionId": "v3",
    "PolicyDocument": "{ \"Statement\": [ { \"Action\": [\"oss:*\"], \"Effect\": \"Allow\", \"Resource\": [\"acs:oss:*:*:*\"], \"Version\": \"1\" } ] }",
    "CreateDate": "2015-01-23T12:33:18Z"
  }
}
```

```
}
}
```

错误码

访问[错误中心](#)查看更多错误码。

7.7 DeletePolicyVersion

调用DeletePolicyVersion接口删除指定的权限策略的某个版本。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	DeletePolicyVersion	系统规定参数。取值：DeletePolicyVersion
PolicyName	String	是	OSS-Administrator	权限策略名称。
VersionId	String	是	v3	指定目标版本的ID。

返回数据

名称	类型	示例值	描述
RequestId	String	9B34724D-54B0-4A51-B34D-4512372FE1BE	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=DeletePolicyVersion
&PolicyName=OSS-Administrator
&VersionId=v3
&<公共请求参数>
```

正常返回示例

XML 格式

```
<DeletePolicyVersionResponse>
  <RequestId>9B34724D-54B0-4A51-B34D-4512372FE1BE</RequestId>
</DeletePolicyVersionResponse>
```

JSON 格式

```
{
  "RequestId": "9B34724D-54B0-4A51-B34D-4512372FE1BE"
}
```

错误码

访问[错误中心](#)查看更多错误码。

7.8 ListPolicyVersions

调用ListPolicyVersions接口列出权限策略版本。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	ListPolicyVersions	系统规定参数。取值：ListPolicyVersions
PolicyName	String	是	OSS-Administrator	权限策略名称。
PolicyType	String	是	Custom	指定权限策略的类型，取值为：System或Custom。

返回数据

名称	类型	示例值	描述
PolicyVersions			权限策略版本信息。
CreateDate	String	2015-02-26T01:25:52Z	创建时间。

名称	类型	示例值	描述
IsDefaultVersion	Boolean	false	是否默认版本。
PolicyDocument	String	{ "Statement": [{ "Action": ["oss:*"], "Effect": "Allow", "Resource": ["acs:oss:*:*:*"], "Version": "1" }	权限策略内容。
VersionId	String	v3	权限策略标识。
RequestId	String	7B8A4E7D-6CFF-471D-84DF-195A7A241ECB	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=ListPolicyVersions
&PolicyName=OSS-Administrator
&PolicyType=Custom
&<公共请求参数>
```

正常返回示例

XML 格式

```
<ListPolicyVersionsResponse>
  <RequestId>7B8A4E7D-6CFF-471D-84DF-195A7A241ECB</RequestId>
  <PolicyVersions>
    <PolicyVersion>
      <VersionId>v3</VersionId>
      <IsDefaultVersion>>false</IsDefaultVersion>
      <CreateDate>2015-01-23T12:33:18Z</CreateDate>
      <PolicyDocument>{ "Statement": [ { "Action": ["oss
:*"], "Effect": "Allow", "Resource": ["acs:oss:*:*:*"], "Version": "
1"}</PolicyDocument>
    </PolicyVersion>
    <PolicyVersion>
      <VersionId>v5</VersionId>
      <IsDefaultVersion>>true</IsDefaultVersion>
      <CreateDate>2015-02-26T01:25:52Z</CreateDate>
      <PolicyDocument>{ "Statement": [ { "Action": ["oss
:*"], "Effect": "Allow", "Resource": ["acs:oss:*:*:*"], "Version": "
1"}</PolicyDocument>
    </PolicyVersion>
  </PolicyVersions>
```



```
</ListPolicyVersionsResponse>
```

JSON 格式

```
{
  "RequestId": "7B8A4E7D-6CFF-471D-84DF-195A7A241ECB",
  "PolicyVersions": {
    "PolicyVersion": [
      {
        "IsDefaultVersion": false,
        "VersionId": "v3",
        "PolicyDocument": "{ \"Statement\": [{ \"Action\": [\"oss:*\"], \"Effect\": \"Allow\", \"Resource\": [\"acs:oss:*:*:*\"]}], \"Version\": \"1\"}",
        "CreateDate": "2015-01-23T12:33:18Z"
      },
      {
        "IsDefaultVersion": true,
        "VersionId": "v5",
        "PolicyDocument": "{ \"Statement\": [{ \"Action\": [\"oss:*\"], \"Effect\": \"Allow\", \"Resource\": [\"acs:oss:*:*:*\"]}], \"Version\": \"1\"}",
        "CreateDate": "2015-02-26T01:25:52Z"
      }
    ]
  }
}
```

错误码

访问[错误中心](#)查看更多错误码。

7.9 SetDefaultPolicyVersion

调用SetDefaultPolicyVersion接口设置权限策略默认版本。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	SetDefaultPolicyVersion	系统规定参数。取值：SetDefaultPolicyVersion
PolicyName	String	是	OSS-Administrator	指定权限策略名称。
VersionId	String	是	v2	新默认版本的ID。

返回数据

名称	类型	示例值	描述
RequestId	String	9B34724D-54B0-4A51-B34D-4512372FE1BE	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=SetDefaultPolicyVersion
&PolicyName=OSS-Administrator
&VersionId=v2
&<公共请求参数>
```

正常返回示例

XML 格式

```
<SetDefaultPolicyVersionResponse>
  <RequestId>9B34724D-54B0-4A51-B34D-4512372FE1BE</RequestId>
</SetDefaultPolicyVersionResponse>
```

JSON 格式

```
{
  "RequestId": "9B34724D-54B0-4A51-B34D-4512372FE1BE"
}
```

错误码

访问[错误中心](#)查看更多错误码。

7.10 AttachPolicyToUser

调用AttachPolicyToUser接口为指定用户添加权限。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	AttachPolicyToUser	系统规定参数。取值: AttachPolicyToUser
PolicyName	String	是	OSS-Administrator	指定权限策略名称。
PolicyType	String	是	Custom	指定权限策略的类型, 取值为: System或Custom。
UserName	String	是	zhangq****	指定用户名。

返回数据

名称	类型	示例值	描述
RequestId	String	697852FB-50D7-44D9-9774-530C31EAC572	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=AttachPolicyToUser
&PolicyType=Custom
&PolicyName=OSS-Administrator
&UserName=zhangq****
&<公共请求参数>
```

正常返回示例

XML 格式

```
<AttachPolicyToUserResponse>
  <RequestId>697852FB-50D7-44D9-9774-530C31EAC572</RequestId>
</AttachPolicyToUserResponse>
```

JSON 格式

```
{
  "RequestId": "697852FB-50D7-44D9-9774-530C31EAC572"
```

```
}

```

错误码

访问[错误中心](#)查看更多错误码。

7.11 DetachPolicyFromUser

调用DetachPolicyFromUser接口为用户撤销指定的权限。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	DetachPolicyFromUser	系统规定参数。取值：DetachPolicyFromUser
PolicyName	String	是	OSS-Administrator	指定权限策略名称。
PolicyType	String	是	Custom	指定权限策略的类型，取值为：System或Custom。
UserName	String	是	zhangq****	指定用户名。

返回数据

名称	类型	示例值	描述
RequestId	String	697852FB-50D7-44D9-9774-530C31EAC572	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=DetachPolicyFromUser
&PolicyType=Custom
&PolicyName=OSS-Administrator
&UserName=zhangq****
```

&<公共请求参数>

正常返回示例

XML 格式

```
<DetachPolicyFromUserResponse>
  <RequestId>697852FB-50D7-44D9-9774-530C31EAC572</RequestId>
</DetachPolicyFromUserResponse>
```

JSON 格式

```
{
  "RequestId": "697852FB-50D7-44D9-9774-530C31EAC572"
}
```

错误码

访问[错误中心](#)查看更多错误码。

7.12 AttachPolicyToGroup

调用AttachPolicyToGroup接口为指定用户组添加权限。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	AttachPolicyToGroup	系统规定参数。取值：AttachPolicyToGroup
GroupName	String	是	dev	指定用户组名称。
PolicyName	String	是	OSS-Administrator	指定权限策略名称。
PolicyType	String	是	Custom	指定权限策略的类型，取值为：System或Custom。

返回数据

名称	类型	示例值	描述
RequestId	String	697852FB-50D7-44D9-9774-530C31EAC572	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=AttachPolicyToGroup
&PolicyType=Custom
&PolicyName=OSS-Administrator
&GroupName=dev
&<公共请求参数>
```

正常返回示例

XML 格式

```
<AttachPolicyToGroupResponse>
  <RequestId>697852FB-50D7-44D9-9774-530C31EAC572</RequestId>
</AttachPolicyToGroupResponse>
```

JSON 格式

```
{
  "RequestId": "697852FB-50D7-44D9-9774-530C31EAC572"
}
```

错误码

访问[错误中心](#)查看更多错误码。

7.13 DetachPolicyFromGroup

调用DetachPolicyFromGroup接口为用户组撤销指定的权限。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	DetachPolicyFromGroup	系统规定参数。取值: DetachPolicyFromGroup
GroupName	String	是	dev	指定用户组名称。
PolicyName	String	是	OSS-Administrator	指定权限策略名称。
PolicyType	String	是	Custom	指定权限策略的类型, 取值为: System或Custom。

返回数据

名称	类型	示例值	描述
RequestId	String	697852FB-50D7-44D9-9774-530C31EAC572	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=DetachPolicyFromGroup
&PolicyType=Custom
&PolicyName=OSS-Administrator
&GroupName=dev
&<公共请求参数>
```

正常返回示例

XML 格式

```
<DetachPolicyFromGroupResponse>
  <RequestId>697852FB-50D7-44D9-9774-530C31EAC572</RequestId>
</DetachPolicyFromGroupResponse>
```

JSON 格式

```
{
  "RequestId": "697852FB-50D7-44D9-9774-530C31EAC572"
```

```
}
}
```

错误码

访问[错误中心](#)查看更多错误码。

7.14 AttachPolicyToRole

调用AttachPolicyToRole接口为指定角色添加权限。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	AttachPolicyToRole	系统规定参数。取值：AttachPolicyToRole
PolicyName	String	是	OSS-Administrator	指定权限策略名称。
PolicyType	String	是	Custom	指定权限策略的类型，取值为：System或Custom。
RoleName	String	是	OSSAdminRole	指定角色名。

返回数据

名称	类型	示例值	描述
RequestId	String	697852FB-50D7-44D9-9774-530C31EAC572	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=AttachPolicyToRole
&PolicyType=Custom
&PolicyName=OSS-Administrator
&RoleName=OSSAdminRole
```


&<公共请求参数>

正常返回示例

XML 格式

```
<AttachPolicyToRoleResponse>
  <RequestId>697852FB-50D7-44D9-9774-530C31EAC572</RequestId>
</AttachPolicyToRoleResponse>
```

JSON 格式

```
{
  "RequestId": "697852FB-50D7-44D9-9774-530C31EAC572"
}
```

错误码

访问[错误中心](#)查看更多错误码。

7.15 DetachPolicyFromRole

调用DetachPolicyFromRole接口为角色撤销指定的权限。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	DetachPolicyFromRole	系统规定参数。取值：DetachPolicyFromRole
PolicyName	String	是	OSS-Administrator	权限策略名称。
PolicyType	String	是	Custom	指定权限策略的类型，取值为：System或Custom。
RoleName	String	是	OSSAdminRole	指定角色名。

返回数据

名称	类型	示例值	描述
RequestId	String	697852FB-50D7-44D9-9774-530C31EAC572	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=DetachPolicyFromRole
&PolicyType=Custom
&PolicyName=OSS-Administrator
&RoleName=OSSAdminRole
&<公共请求参数>
```

正常返回示例

XML 格式

```
<DetachPolicyFromRoleResponse>
  <RequestId>697852FB-50D7-44D9-9774-530C31EAC572</RequestId>
</DetachPolicyFromRoleResponse>
```

JSON 格式

```
{
  "RequestId": "697852FB-50D7-44D9-9774-530C31EAC572"
}
```

错误码

访问[错误中心](#)查看更多错误码。

7.16 ListPoliciesForUser

调用ListPoliciesForUser接口列出指定用户的权限策略。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	ListPoliciesForUser	系统规定参数。取值: ListPoliciesForUser
UserName	String	是	zhangq****	指定用户名。

返回数据

名称	类型	示例值	描述
Policies			权限策略信息。
AttachDate	String	2015-01-23T12:33:18Z	授权时间。
DefaultVersion	String	v1	默认版本。
Description	String	OSS管理员权限	权限策略描述。
PolicyName	String	OSS-Administrator	权限策略名称。
PolicyType	String	Custom	权限策略类型。
RequestId	String	7B8A4E7D-6CFF-471D-84DF-195A7A241ECB	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=ListPoliciesForUser
&UserName=zhangq****
&<公共请求参数>
```

正常返回示例

XML 格式

```
<ListPoliciesForUserResponse>
  <RequestId>7B8A4E7D-6CFF-471D-84DF-195A7A241ECB</RequestId>
  <Policies>
    <Policy>
      <PolicyName>OSS-Administrator</PolicyName>
```

```
<PolicyType>Custom</PolicyType>
<Description>OSS管理员权限</Description>
<DefaultVersion>v1</DefaultVersion>
<AttachDate>2015-01-23T12:33:18Z</AttachDate>
</Policy>
<Policy>
  <PolicyName>ReadOnlyAccess</PolicyName>
  <PolicyType>System</PolicyType>
  <Description>只读权限</Description>
  <DefaultVersion>v1</DefaultVersion>
  <AttachDate>2015-02-18T17:22:08Z</AttachDate>
</Policy>
</Policies>
</ListPoliciesForUserResponse>
```

JSON 格式

```
{
  "Policies":{
    "Policy":[
      {
        "Description":"OSS管理员权限",
        "PolicyName":"OSS-Administrator",
        "AttachDate":"2015-01-23T12:33:18Z",
        "DefaultVersion":"v1",
        "PolicyType":"Custom"
      },
      {
        "Description":"只读权限",
        "PolicyName":"ReadOnlyAccess",
        "AttachDate":"2015-02-18T17:22:08Z",
        "DefaultVersion":"v1",
        "PolicyType":"System"
      }
    ]
  },
  "RequestId":"7B8A4E7D-6CFF-471D-84DF-195A7A241ECB"
}
```

错误码

访问[错误中心](#)查看更多错误码。

7.17 ListEntitiesForPolicy

调用ListEntitiesForPolicy接口列出引用权限策略的实体。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	ListEntitiesForPolicy	系统规定参数。取值: ListEntitiesForPolicy
PolicyName	String	是	OSS-Administrator	权限策略名称。
PolicyType	String	是	Custom	指定权限策略的类型, 取值为: System或Custom。

返回数据

名称	类型	示例值	描述
Users			用户信息。
AttachDate	String	2015-01-23T12:33:18Z	授权时间。
DisplayName	String	张*	显示名称。
UserId	String	122748924538****	用户唯一标识。
UserName	String	zhangq****	用户名称。
Groups			用户组信息。
AttachDate	String	2015-02-18T17:22:08Z	授权时间。
Comments	String	测试团队	备注信息。
GroupName	String	QA-Team	组名称。
Roles			角色信息。
Arn	String	acs:ram::123456789012****:role/ECSAdmin	角色的资源描述符。

名称	类型	示例值	描述
AttachDate	String	2015-01-23T12:33:18Z	授权时间。
Description	String	ECS管理角色	角色的文字描述。
RoleId	String	122748924538****	角色ID。
RoleName	String	ECSAdmin	角色名称。
RequestId	String	7B8A4E7D-6CFF-471D-84DF-195A7A241ECB	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=ListEntitiesForPolicy
&PolicyName=OSS-Administrator
&PolicyType=Custom
&<公共请求参数>
```

正常返回示例

XML 格式

```
<ListEntitiesForPolicyResponse>
  <RequestId>7B8A4E7D-6CFF-471D-84DF-195A7A241ECB</RequestId>
  <Groups>
    <Group>
      <GroupName>QA-Team</GroupName>
      <Comments>测试团队</Comments>
      <AttachDate>2015-01-23T12:33:18Z</AttachDate>
    </Group>
    <Group>
      <GroupName>Dev-Team</GroupName>
      <Comments>开发团队</Comments>
      <AttachDate>2015-02-18T17:22:08Z</AttachDate>
    </Group>
  </Groups>
  <Users>
    <User>
      <UserId>122748924538****</UserId>
      <UserName>zhangq****</UserName>
      <DisplayName>张*</DisplayName>
      <AttachDate>2015-01-23T12:33:18Z</AttachDate>
    </User>
    <User>
      <UserId>140649822472****</UserId>
      <UserName>li****</UserName>
      <DisplayName>李*</DisplayName>
      <AttachDate>2015-02-18T17:22:08Z</AttachDate>
    </User>
  </Users>
</ListEntitiesForPolicyResponse>
```

```

        </User>
    </Users>
    <Roles>
        <Role>
            <RoleId>122748924538****</RoleId>
            <RoleName>ECSAdmin</RoleName>
            <Arn>acs:ram::123456789012****:role/ECSAdmin</Arn>
            <Description>ECS管理角色</Description>
            <AttachDate>2015-01-23T12:33:18Z</AttachDate>
        </Role>
        <Role>
            <RoleId>140649822472****</RoleId>
            <RoleName>OSSReadOnlyAccess</RoleName>
            <Description>OSS只读访问角色</Description>
            <Arn>acs:ram::123456789012****:role/OSSReadOnlyAccess</Arn>
            <AttachDate>2015-01-23T12:33:18Z</AttachDate>
        </Role>
    </Roles>
</ListEntitiesForPolicyResponse>

```

JSON 格式

```

{
  "Groups":{
    "Group":[
      {
        "Comments":"测试团队",
        "GroupName":"QA-Team",
        "AttachDate":"2015-01-23T12:33:18Z"
      },
      {
        "Comments":"开发团队",
        "GroupName":"Dev-Team",
        "AttachDate":"2015-02-18T17:22:08Z"
      }
    ]
  },
  "Users":{
    "User":[
      {
        "UserName":"zhangq****",
        "UserId":"122748924538****",
        "DisplayName":"张*",
        "AttachDate":"2015-01-23T12:33:18Z"
      },
      {
        "UserName":"li****",
        "UserId":"140649822472****",
        "DisplayName":"李*",
        "AttachDate":"2015-02-18T17:22:08Z"
      }
    ]
  },
  "RequestId":"7B8A4E7D-6CFF-471D-84DF-195A7A241ECB",
  "Roles":{
    "Role":[
      {
        "RoleName":"ECSAdmin",
        "Description":"ECS管理角色",
        "Arn":"acs:ram::123456789012****:role/ECSAdmin",
        "AttachDate":"2015-01-23T12:33:18Z",
        "RoleId":"122748924538****"
      }
    ]
  }
}

```

```

    },
    {
      "RoleName": "OSSReadOnlyAccess",
      "Description": "OSS只读访问角色",
      "Arn": "acs:ram::123456789012****:role/OSSReadOnlyAccess",
      "AttachDate": "2015-02-18T17:22:08Z",
      "RoleId": "140649822472****"
    }
  ]
}
}
}

```

错误码

访问[错误中心](#)查看更多错误码。

7.18 ListPoliciesForGroup

调用ListPoliciesForGroup接口列出用户组的权限策略。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	ListPoliciesForGroup	系统规定参数。取值：ListPoliciesForGroup
GroupName	String	是	dev	指定用户组名称。

返回数据

名称	类型	示例值	描述
Policies			权限策略信息。
AttachDate	String	2015-01-23T12:33:18Z	授权时间。
DefaultVersion	String	v1	默认版本。
Description	String	OSS管理员权限	权限策略描述。
PolicyName	String	OSS-Administrator	权限策略名称。

名称	类型	示例值	描述
PolicyType	String	Custom	权限策略类型。
RequestId	String	7B8A4E7D-6CFF-471D-84DF-195A7A241ECB	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=ListPoliciesForGroup
&GroupName=dev
&<公共请求参数>
```

正常返回示例

XML 格式

```
<ListPoliciesForGroupResponse>
  <RequestId>7B8A4E7D-6CFF-471D-84DF-195A7A241ECB</RequestId>
  <Policies>
    <Policy>
      <PolicyName>OSS-Administrator</PolicyName>
      <PolicyType>Custom</PolicyType>
      <Description>OSS管理员权限</Description>
      <DefaultVersion>v1</DefaultVersion>
      <AttachDate>2015-01-23T12:33:18Z</AttachDate>
    </Policy>
    <Policy>
      <PolicyName>ReadOnlyAccess</PolicyName>
      <PolicyType>System</PolicyType>
      <Description>只读权限</Description>
      <DefaultVersion>v1</DefaultVersion>
      <AttachDate>2015-02-18T17:22:08Z</AttachDate>
    </Policy>
  </Policies>
</ListPoliciesForGroupResponse>
```

JSON 格式

```
{
  "Policies":{
    "Policy":[
      {
        "Description":"OSS管理员权限",
        "PolicyName":"OSS-Administrator",
        "AttachDate":"2015-01-23T12:33:18Z",
        "DefaultVersion":"v1",
        "PolicyType":"Custom"
      },
      {
        "Description":"只读权限",
        "PolicyName":"ReadOnlyAccess",
        "AttachDate":"2015-02-18T17:22:08Z",
```

```

    "DefaultVersion": "v1",
    "PolicyType": "System"
  }
},
"RequestId": "7B8A4E7D-6CFF-471D-84DF-195A7A241ECB"
}

```

错误码

访问[错误中心](#)查看更多错误码。

7.19 ListPoliciesForRole

调用ListPoliciesForRole接口列出角色的权限策略。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	ListPoliciesForRole	系统规定参数。取值：ListPoliciesForRole
RoleName	String	是	AdminRole	指定角色名。

返回数据

名称	类型	示例值	描述
Policies			权限策略信息。
AttachDate	String	2015-01-23T12:33:18Z	授权时间。
DefaultVersion	String	v1	默认版本。
Description	String	OSS管理员权限	权限策略描述。
PolicyName	String	OSS-Administrator	权限策略名称。
PolicyType	String	Custom	权限策略类型。

名称	类型	示例值	描述
RequestId	String	7B8A4E7D-6CFF-471D-84DF-195A7A241ECB	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=ListPoliciesForRole
&RoleName=AdminRole
&<公共请求参数>
```

正常返回示例

XML 格式

```
<ListPoliciesForRoleResponse>
  <RequestId>7B8A4E7D-6CFF-471D-84DF-195A7A241ECB</RequestId>
  <Policies>
    <Policy>
      <PolicyName>OSS-Administrator</PolicyName>
      <PolicyType>Custom</PolicyType>
      <Description>OSS管理员权限</Description>
      <DefaultVersion>v1</DefaultVersion>
      <AttachDate>2015-01-23T12:33:18Z</AttachDate>
    </Policy>
    <Policy>
      <PolicyName>ReadOnlyAccess</PolicyName>
      <PolicyType>System</PolicyType>
      <Description>只读权限</Description>
      <DefaultVersion>v1</DefaultVersion>
      <AttachDate>2015-02-18T17:22:08Z</AttachDate>
    </Policy>
  </Policies>
</ListPoliciesForRoleResponse>
```

JSON 格式

```
{
  "Policies":{
    "Policy":[
      {
        "Description":"OSS管理员权限",
        "PolicyName":"OSS-Administrator",
        "AttachDate":"2015-01-23T12:33:18Z",
        "DefaultVersion":"v1",
        "PolicyType":"Custom"
      },
      {
        "Description":"只读权限",
        "PolicyName":"ReadOnlyAccess",
        "AttachDate":"2015-02-18T17:22:08Z",
        "DefaultVersion":"v1",
        "PolicyType":"System"
      }
    ]
  }
}
```

```
]
},
"RequestId": "7B8A4E7D-6CFF-471D-84DF-195A7A241ECB"
}
```

错误码

访问[错误中心](#)查看更多错误码。

8 安全设置接口

8.1 SetAccountAlias

调用SetAccountAlias接口设置云账号别名。

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。

请求参数

名称	类型	是否必选	示例值	描述
AccountAlias	String	是	myalias	指定云账号的别名, 长度限制为3~32个字符。 限制: <code>^[a-z0-9]([a-z0-9] -(?!-))*[a-z0-9]?\$。</code>
Action	String	是	SetAccountAlias	系统规定参数。取值: SetAccountAlias

返回数据

名称	类型	示例值	描述
RequestId	String	04F0F334-1335-436C-A1D7-6C044FE73368	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=SetAccountAlias
&AccountAlias=myalias
&<公共请求参数>
```

正常返回示例

XML 格式

```
<SetAccountAliasResponse>
  <RequestId>04F0F334-1335-436C-A1D7-6C044FE73368</RequestId>
</SetAccountAliasResponse>
```

JSON 格式

```
{
  "RequestId": "04F0F334-1335-436C-A1D7-6C044FE73368"
}
```

错误码

访问[错误中心](#)查看更多错误码。

8.2 GetAccountAlias

调用GetAccountAlias接口查看云账号别名。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	GetAccountAlias	系统规定参数。取值：GetAccountAlias

返回数据

名称	类型	示例值	描述
AccountAlias	String	myalias	云账号别名。
RequestId	String	04F0F334-1335-436C-A1D7-6C044FE73368	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=GetAccountAlias
```

&<公共请求参数>

正常返回示例

XML 格式

```
<GetAccountAliasResponse>
  <RequestId>04F0F334-1335-436C-A1D7-6C044FE73368</RequestId>
  <AccountAlias>myalias</AccountAlias>
</GetAccountAliasResponse>
```

JSON 格式

```
{
  "RequestId": "04F0F334-1335-436C-A1D7-6C044FE73368",
  "AccountAlias": "myalias"
}
```

错误码

访问[错误中心](#)查看更多错误码。

8.3 ClearAccountAlias

调用ClearAccountAlias接口清除云账号别名。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	ClearAccountAlias	系统规定参数。取值：ClearAccountAlias

返回数据

名称	类型	示例值	描述
RequestId	String	04F0F334-1335-436C-A1D7-6C044FE73368	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=ClearAccountAlias
&<公共请求参数>
```

正常返回示例

XML 格式

```
<ClearAccountAliasResponse>
  <RequestId>04F0F334-1335-436C-A1D7-6C044FE73368</RequestId>
</ClearAccountAliasResponse>
```

JSON 格式

```
{
  "RequestId": "04F0F334-1335-436C-A1D7-6C044FE73368"
}
```

错误码

访问[错误中心](#)查看更多错误码。

8.4 SetPasswordPolicy

调用SetPasswordPolicy接口设置RAM用户密码强度等策略信息。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	SetPasswordPolicy	系统规定参数。取值：SetPasswordPolicy

名称	类型	是否必选	示例值	描述
HardExpiry	Boolean	否	false	<p>密码是否过期。</p> <p>取值为true或false，如果未指定此参数的值，系统默认为false。</p> <ul style="list-style-type: none"> · 如果取值为true，此时必须由主账号重置密码后，RAM用户才能正常登录。 · 如果取值为false，此时RAM用户可以在密码过期后自行更改密码，并继续以用户身份登录。
MaxLoginAttemps	Integer	否	5	一小时内使用错误密码尝试登录最大次数（重置密码可清除尝试登录计数）。
MaxPasswordAge	Integer	否	0	密码有效期，单位是天（重置密码将重置密码过期时间）。0表示不启用密码过期策略，默认不启用。
MinimumPasswordLength	Integer	否	12	最小密码长度。
PasswordReusePrevention	Integer	否	0	禁止使用前N次密码。0表示不启用历史密码检查策略，默认不启用。
RequireLowercaseCharacters	Boolean	否	true	必须含有小写字母。
RequireNumbers	Boolean	否	true	必须含有数字。
RequireSymbols	Boolean	否	true	必须含有特殊字符。
RequireUppercaseCharacters	Boolean	否	true	必须含有大写字母。

返回数据

名称	类型	示例值	描述
PasswordPolicy			密码策略。
HardExpiry	Boolean	false	密码是否过期。 取值为true或false，如果未指定此参数的值，系统默认为false。 · 如果取值为true，此时必须由主账号重置密码后，RAM用户才能正常登录。 · 如果取值为false，此时RAM用户可以在密码过期后自行更改密码，并继续以用户身份登录。
MaxLoginAttempts	Integer	5	一小时内使用错误密码尝试登录最大次数（重置密码可清除尝试登录计数）。
MaxPasswordAge	Integer	0	密码有效期，单位是天（重置密码将重置密码过期时间）。0表示不启用密码过期策略，默认不启用。
MinimumPasswordLength	Integer	12	最小密码长度。
PasswordReusePrevention	Integer	0	禁止使用前N次密码。0表示不启用历史密码检查策略，默认不启用。
RequireLowercaseCharacters	Boolean	true	必须包含小写字母。
RequireNumbers	Boolean	true	必须包含数字。
RequireSymbols	Boolean	true	必须包含字符。
RequireUppercaseCharacters	Boolean	true	必须包含大写字母。

名称	类型	示例值	描述
RequestId	String	04F0F334-1335-436C-A1D7-6C044FE73368	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=SetPasswordPolicy
&MinimumPasswordLength=12
&RequireLowercaseCharacters=true
&RequireUppercaseCharacters=true
&RequireNumbers=true
&RequireSymbols=true
&<公共请求参数>
```

正常返回示例

XML 格式

```
<SetPasswordPolicyResponse>
  <RequestId>04F0F334-1335-436C-A1D7-6C044FE73368</RequestId>
  <PasswordPolicy>
    <HardExpiry>>false</HardExpiry>
    <MaxLoginAttempts>5</MaxLoginAttempts>
    <MaxPasswordAge>0</MaxPasswordAge>
    <PasswordReusePrevention>0</PasswordReusePrevention>
    <MinimumPasswordLength>12</MinimumPasswordLength>
    <RequireLowercaseCharacters>true</RequireLowercaseChar
acters>
    <RequireUppercaseCharacters>true</RequireUppercaseChar
acters>
    <RequireNumbers>true</RequireNumbers>
    <RequireSymbols>true</RequireSymbols>
  </PasswordPolicy>
</SetPasswordPolicyResponse>
```

JSON 格式

```
{
  "RequestId": "04F0F334-1335-436C-A1D7-6C044FE73368",
  "PasswordPolicy": {
    "RequireUppercaseCharacters": true,
    "MaxPasswordAge": 0,
    "HardExpiry": false,
    "RequireNumbers": true,
    "RequireSymbols": true,
    "MaxLoginAttempts": 5,
    "PasswordReusePrevention": 0,
    "RequireLowercaseCharacters": true,
    "MinimumPasswordLength": 12
  }
}
```

```
}

```

错误码

访问[错误中心](#)查看更多错误码。

8.5 GetPasswordPolicy

调用GetPasswordPolicy接口获取RAM用户密码强度等策略信息。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	GetPasswor dPolicy	系统规定参数。取值：GetPasswor dPolicy

返回数据

名称	类型	示例值	描述
PasswordPo licy			密码策略。
HardExpiry	Boolean	false	密码是否过期。 取值为true或false，如果未指定此参数的值，系统默认为false。 <ul style="list-style-type: none"> 如果取值为true，此时必须由主账号重置密码后，RAM用户才能正常登录。 如果取值为false，此时RAM用户可以在密码过期后自行更改密码，并继续以用户身份登录。
MaxLoginAt temps	Integer	5	一小时内使用错误密码尝试登录最大次数（重置密码可清除尝试登录计数）。

名称	类型	示例值	描述
MaxPasswordAge	Integer	0	密码有效期，单位为天（重置密码将重置密码过期时间）。0表示不启用密码过期策略，默认不启用。
MinimumPasswordLength	Integer	12	最小密码长度。
PasswordReusePrevention	Integer	0	禁止使用前N次密码。0表示不启用历史密码检查策略，默认不启用。
RequireLowercaseCharacters	Boolean	true	必须包含小写字母。
RequireNumbers	Boolean	true	必须包含数字。
RequireSymbols	Boolean	true	必须包含字符。
RequireUppercaseCharacters	Boolean	true	必须包含大写字母。
RequestId	String	04F0F334-1335-436C-A1D7-6C044FE73368	请求ID。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=GetPasswordPolicy
&AccountAlias=myalias
&<公共请求参数>
```

正常返回示例

XML 格式

```
<GetPasswordPolicyResponse>
  <RequestId>04F0F334-1335-436C-A1D7-6C044FE73368</RequestId>
  <PasswordPolicy>
    <HardExpiry>>false</HardExpiry>
    <MaxLoginAttempts>5</MaxLoginAttempts>
    <MaxPasswordAge>0</MaxPasswordAge>
    <PasswordReusePrevention>0</PasswordReusePrevention>
    <MinimumPasswordLength>12</MinimumPasswordLength>
```

```

    <RequireLowercaseCharacters>true</RequireLowercaseChar
acters>
    <RequireUppercaseCharacters>true</RequireUppercaseChar
acters>
    <RequireNumbers>true</RequireNumbers>
    <RequireSymbols>true</RequireSymbols>
  </PasswordPolicy>
</GetPasswordPolicyResponse>

```

JSON 格式

```

{
  "RequestId": "04F0F334-1335-436C-A1D7-6C044FE73368",
  "PasswordPolicy": {
    "RequireUppercaseCharacters": true,
    "MaxPasswordAge": 0,
    "HardExpiry": false,
    "RequireNumbers": true,
    "RequireSymbols": true,
    "MaxLoginAttempts": 5,
    "PasswordReusePrevention": 0,
    "RequireLowercaseCharacters": true,
    "MinimumPasswordLength": 12
  }
}

```

错误码

访问[错误中心](#)查看更多错误码。

8.6 SetSecurityPreference

调用SetSecurityPreference接口设置全局安全首选项。

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。

请求参数

名称	类型	是否必选	示例值	描述
Action	String	是	SetSecurityPreference	系统规定参数。取值：SetSecurityPreference
AllowUserToChangePassword	Boolean	否	true	是否允许用户修改自己的密码。

名称	类型	是否必选	示例值	描述
AllowUserToManageAccessKeys	Boolean	否	false	是否允许用户管理自己的访问密钥。
AllowUserToManageMFADevices	Boolean	否	true	是否允许用户管理自己绑定或解绑多因素认证设备。
AllowUserToManagePublicKeys	Boolean	否	false	是否允许用户管理自己的公钥。
EnableSaveMFATicket	Boolean	否	true	是否允许在用户在登录时保存多因素认证设备验证票据，目前票据有效期是七天。
LoginNetworkMasks	String	否	10.0.0.0/8	登录掩码，默认空字符串，不限制登录IP。
LoginSessionDuration	Integer	否	6	用户登录有效期，单位：小时。

返回数据

名称	类型	示例值	描述
RequestId	String	04F0F334-1335-436C-A1D7-6C044FE73368	请求ID。
SecurityPreference			安全首选项。
AccessKeyPreference			访问密钥首选项。
AllowUserToManageAccessKeys	Boolean	false	是否允许用户管理自己的访问密钥。
LoginProfilePreference			登录首选项。
AllowUserToChangePassword	Boolean	true	是否允许用户修改自己的密码。

名称	类型	示例值	描述
EnableSaveMFATicket	Boolean	true	是否允许在用户在登录时保存多因素认证设备验证票据。
LoginNetworkMasks	String	10.0.0.0/8	登录掩码，默认空字符串，不限制登录IP。
LoginSessionDuration	Integer	6	用户登录有效期，单位：小时。
MFAPreference			多因素认证首选项。
AllowUserToManageMFADevices	Boolean	true	是否允许用户管理自己绑定或解绑多因素认证设备。
PublicKeyPreference			公钥首选项。
AllowUserToManagePublicKeys	Boolean	false	是否允许用户管理自己的公钥。

示例

请求示例

```
https://ram.aliyuncs.com/?Action=SetSecurityPreference
&EnableSaveMFATicket=true
&AllowUserToChangePassword=true
&AllowUserToManageAccessKeys=false
&<公共请求参数>
```

正常返回示例

XML 格式

```
<SetSecurityPreferenceResponse>
  <RequestId>04F0F334-1335-436C-A1D7-6C044FE73368</RequestId>
  <SecurityPreference>
    <LoginProfilePreference>
      <EnableSaveMFATicket>true</EnableSaveMFATicket>
      <AllowUserToChangePassword>true</AllowUserToChangePassword>
    </LoginProfilePreference>
    <LoginNetworkMasks>10.0.0.0/8</LoginNetworkMasks>
    <LoginSessionDuration>6</LoginSessionDuration>
  </SecurityPreference>
  <AccessKeyPreference>
    <AllowUserToManageAccessKeys>false</AllowUserToManageAccessKeys>
  </AccessKeyPreference>
</SetSecurityPreferenceResponse>
```



```
        </AccessKeyPreference>
        <MFAPreference>
          <AllowUserToManageMFADevices>>false</AllowUserT
oManageMFADevices>
        </MFAPreference>
        <PublicKeyPreference>
          <AllowUserToManagePublicKeys>>false</AllowUserT
oManagePublicKeys>
        </PublicKeyPreference>
      </SecurityPreference>
    </SetSecurityPreferenceResponse>
```

JSON 格式

```
{
  "RequestId": "04F0F334-1335-436C-A1D7-6C044FE73368",
  "SecurityPreference": {
    "LoginProfilePreference": {
      "LoginSessionDuration": 6,
      "LoginNetworkMasks": "10.0.0.0/8",
      "EnableSaveMFATicket": true,
      "AllowUserToChangePassword": true
    },
    "AccessKeyPreference": {
      "AllowUserToManageAccessKeys": false
    },
    "PublicKeyPreference": {
      "AllowUserToManagePublicKeys": false
    },
    "MFAPreference": {
      "AllowUserToManageMFADevices": true
    }
  }
}
```

错误码

访问[错误中心](#)查看更多错误码。

9 (隐藏) 旧版 API 参考 (RAM)

9.1 简介

9.1.1 RAM 简介

什么是 RAM?

RAM (Resource Access Management) 是阿里云提供的资源访问控制服务。通过 RAM, 您可以集中管理您的用户 (比如员工、系统或应用程序), 以及控制用户可以访问您名下哪些资源的权限。

RAM 要解决的问题

如果您购买了云资源, 您的组织里有多个用户需要使用这些云资源, 这些用户只能共享使用您的云账号密钥, 那么就存在两个问题:

1. 密钥由多人共享, 泄露的风险很高;
2. 无法控制特定用户能访问哪些资源的权限。

RAM 解决问题的方法

- RAM 允许在您的云账户下创建多个 RAM 用户, 并给这些用户分配资源操作权限。
- RAM 用户不允许拥有资源, 没有独立的计量计费, 这些用户由您的云账户统一控制和付费。
- 您可以给每个 RAM 用户创建单独的登录密码或访问密钥, 但在默认条件下, 这些用户并没有任何操作权限。RAM 提供基于访问策略的授权机制, 支持您实现对 RAM 用户的细颗粒度授权。

RAM 功能

RAM 包括下列功能:

- 集中控制 RAM 用户及其密钥: 您可以管理每个用户及其访问密钥, 为用户绑定/解绑多因素认证设备。
- 集中控制 RAM 用户的访问权限: 您可以控制每个用户可以访问您名下哪些资源的操作权限。
- 集中控制 RAM 用户的资源访问方式: 您可以确保用户必须使用安全信道 (如 SSL)、在指定时间、以及在指定的网络环境下请求访问特定的云服务。
- 集中控制云资源: 您可以对用户创建的实例或数据进行集中控制。当用户离开您的组织时, 这些实例或数据不会丢失。
- 统一账单: 您的账户将收到包括所有用户的资源操作所发生的费用的单一账单。

RAM访问点

用于 API 访问的 RAM 接入点: <https://ram.aliyuncs.com>。

支持RAM的云产品

几乎所有的阿里云产品都会与 RAM 集成。目前支持哪些服务与 RAM 集成的信息, 请参考 [#unique_136](#)。

RAM服务费用

RAM 服务是阿里云账户的一个功能, 无需额外付费。您只需为 RAM 用户所使用的其它服务付费。

9.1.2 术语表

术语	中文	说明
Account	账户	这里是指一个云账户
User	用户	这里是指一个账户下的RAM用户
Group	群组、用户组	这里是指一个账户下的RAM用户组
Policy	授权策略	用来描述授权策略的一种描述语言
AK (Access Key)	访问密钥	访问密钥由AccessKeyID和AccessKeySecret组成, 用于云服务API请求的身份认证
MFA (Multi-Factor Authentication)	多因素认证	您可以给账户或RAM用户绑定MFA设备来增强安全性

9.2 错误代码表

本文描述了 RAM API 错误相关信息。详细信息请参考[API 错误中心](#)。

HTTP Status 400

InvalidParameter

InvalidParameter.UserName.InvalidChars

- HTTP Status: 400
- Error Message: The parameter - "UserName" contains invalid chars.

- 解决方案: 修改 “UserName” 的值, 允许输入大小写英文字母、数字、“.”、“_”或“-”。

InvalidParameter.UserName.Length

- HTTP Status: 400
- Error Message: The parameter - "UserName" beyond the length limit.
- 解决方案: 修改 “UserName” 的长度, 长度范围为: 1-64 个字符。

InvalidParameter.NewUserName.InvalidChars

- HTTP Status: 400
- Error Message: The parameter - "NewUserName" contains invalid chars.
- 解决方案: 修改 “NewUserName” 的值, 允许输入大小写英文字母、数字、“.”、“_”或“-”。

InvalidParameter.NewUserName.Length

- HTTP Status: 400
- Error Message: The parameter - "NewUserName" beyond the length limit.
- 解决方案: 修改 “NewUserName” 的长度, 长度范围为: 1-64 个字符。

InvalidParameter.DisplayName.InvalidChars

- HTTP Status: 400
- Error Message: The parameter - "DisplayName" contains invalid chars.
- 解决方案: 修改 “DisplayName” 的值, 允许输入英文字母、数字、“@”、“.”、“_”或“-”。

InvalidParameter.DisplayName.Length

- HTTP Status: 400
- Error Message: The parameter - "DisplayName" beyond the length limit.
- 解决方案: 修改 “DisplayName” 的长度, 长度范围为: 1-128 个字符或汉字。

InvalidParameter.NewDisplayName.InvalidChars

- HTTP Status: 400
- Error Message: The parameter - "NewDisplayName" contains invalid chars.
- 解决方案: 修改 “NewDisplayName” 的值, 允许输入英文字母、数字、“@”、“.”、“_”或“-”。

InvalidParameter.NewDisplayName.Length

- HTTP Status: 400
- Error Message: The parameter - "NewDisplayName" beyond the length limit.
- 解决方案: 修改 “NewDisplayName” 的长度, 长度范围为: 1-128 个字符或汉字。

InvalidParameter.MobilePhone.Format

- HTTP Status: 400
- Error Message: The format of the parameter - "MobilePhone" is incorrect.
- 解决方案: 修改 “MobilePhone” 的值, 输入正确的手机号码。

InvalidParameter.NewMobilePhone.Format

- HTTP Status: 400
- Error Message: The format of the parameter - "NewMobilePhone" is incorrect.
- 解决方案: 修改 “NewMobilePhone” 的值, 输入正确的手机号码。

InvalidParameter.Email.Format

- HTTP Status: 400
- Error Message: The format of the parameter - "Email" is incorrect.
- 解决方案: 修改 “Email” 的值, 输入正确的邮箱地址。

InvalidParameter.NewEmail.Format

- HTTP Status: 400
- Error Message: The format of the parameter - "NewEmail" is incorrect.
- 解决方案: 修改 “NewEmail” 的值, 输入正确的邮箱地址。

InvalidParameter.Comments.Length

- HTTP Status: 400
- Error Message: The parameter - "Comments" beyond the length limit.
- 解决方案: “Comments” 长度超出限制, 最大长度为 128 个字符。

InvalidParameter.NewComments.Length

- HTTP Status: 400
- Error Message: The parameter - "NewComments" beyond the length limit.
- 解决方案: “NewComments” 长度超出限制, 最大长度为 128 个字符。

InvalidParameter.Password.TooWeak

- HTTP Status: 400

- **Error Message:** The parameter - "Password" is not compliant with the password policy.
- **解决方案:** 所设置的密码不符合密码强度设置。

InvalidParameter.VirtualMFADeviceName.InvalidChars

- **HTTP Status:** 400
- **Error Message:** The parameter - "VirtualMFADeviceName" contains invalid chars.
- **解决方案:** 虚拟 MFA 设备名字包含不合法字符, 请输入正确的虚拟 MFA 设备名字。

InvalidParameter.VirtualMFADeviceName.Length

- **HTTP Status:** 400
- **Error Message:** The parameter - "VirtualMFADeviceName" beyond the length limit .
- **解决方案:** 虚拟 MFA 设备名字超过长度限制, 最大长度为 64 个字符。

InvalidParameter.GroupName.InvalidChars

- **HTTP Status:** 400
- **Error Message:** The parameter - "GroupName" contains invalid chars.
- **解决方案:** 修改 “GroupName” 值, 允许英文字母、数字, 或 “-”。

InvalidParameter.GroupName.Length

- **HTTP Status:** 400
- **Error Message:** The parameter - "GroupName" beyond the length limit.
- **解决方案:** 修改 “GroupName” 的值, 长度范围为: 1-64 个字符。

InvalidParameter.NewGroupName.InvalidChars

- **HTTP Status:** 400
- **Error Message:** The parameter - "NewGroupName" contains invalid chars.
- **解决方案:** 修改 “NewGroupName” 的值, 允许英文字母、数字, 或 “-”。

InvalidParameter.NewGroupName.Length

- **HTTP Status:** 400
- **Error Message:** The parameter - "NewGroupName" beyond the length limit.
- **解决方案:** 修改 “NewGroupName” 的值, 长度范围为: 1-64 个字符。

InvalidParameter.PolicyType

- **HTTP Status:** 400

- Error Message: The parameter - "PolicyType" is incorrect.
- 解决方案: 修改 “PolicyType” 的值, 取值为 System 或 Custom。

InvalidParameter.PolicyName.InvalidChars

- HTTP Status: 400
- Error Message: The parameter - "PolicyName" contains invalid chars.
- 解决方案: 修改 “PolicyName” 的值, 允许英文字母、数字, 或 “-”。

InvalidParameter.PolicyName.Length

- HTTP Status: 400
- Error Message: The parameter - "PolicyName" beyond the length limit.
- 解决方案: 修改 “PolicyName” 的值, 长度范围为: 1-128 个字符。

InvalidParameter.PolicyDocument.Length

- HTTP Status: 400
- Error Message: The parameter - "PolicyName" beyond the length limit.
- 解决方案: 修改 “PolicyDocument” 的值, 最大长度为 2048 字节。

InvalidParameter.Description.Length

- HTTP Status: 400
- Error Message: The parameter - "Description" beyond the length limit.
- 解决方案: 修改 “Description” 的值, 最大长度 1024 字节。

InvalidParameter.VersionId.Format

- HTTP Status: 400
- Error Message: The format of the parameter - "VersionId" is incorrect.
- 解决方案: 修改 “VersionId” 的值, 使其与系统中真实存在的授权策略的 VersionId 保持一致。

InvalidParameter.AccountAlias.Format

- HTTP Status: 400
- Error Message: The format of parameter - "AccountAlias" is incorrect.
- 解决方案: 修改 “AccountAlias” 的值, 仅可以包含小写字母, 数字以及中划线, 不能以中划线作为开头或结尾, 并且不能有两个连续的中划线。

InvalidParameter.AccountAlias.Length

- HTTP Status: 400

- Error Message: The parameter - "AccountAlias" MUST be in the range of 3 and 63.
- 解决方案: 修改 “AccountAlias” 的值, 长度范围为: 3-63 个字符。

InvalidParameter.MinimumPasswordLength.Range

- HTTP Status: 400
- Error Message: The parameter - "MinimumPasswordLength" MUST be in the range of 8 and 32.
- 解决方案: 修改 “MinimumPasswordLength” 的值, 范围在 8-32 之间, 包含 8 和 32。

InvalidParameter.AuthenticationCode.Format

- HTTP Status: 400
- Error Message: The parameter - "AuthenticationCode" MUST be 6 numbers.
- 解决方案: 修改 “AuthenticationCode” 的值, 必须为 6 位数字。

MalformedPolicyDocument

- HTTP Status: 400
- Error Message: {The error details}
- 解决方案: 修改策略内容。

HTTP Status 403

NoPermission

- HTTP Status: 403
- Error Message: You are not authorized to do this action.
- 解决方案: 子用户权限不够, 请给予子用户授权。

Inactive

- HTTP Status: 403
- Error Message: Your ram service is inactive, please activate it.
- 解决方案: 未开通 RAM, 请先开通访问控制 (RAM) 服务。

CheckAuthenticationCodeFail

- HTTP Status: 403
- Error Message: The parameter - "AuthenticationCode" is incorrect.
- 解决方案: 请输入正确的 MFA 安全码。

HTTP Status 404

EntityNotExistEntityNotExist.User

- HTTP Status: 404
- Error Message: The user does not exist.
- 解决方案: 请确认当前子用户在系统中是否存在。

EntityNotExist.User.AccessKey

- HTTP Status: 404
- Error Message: The user access key does not exist.
- 解决方案: 请确认当前 AccessKey 在系统中是否存在。

EntityNotExist.User.LoginProfile

- HTTP Status: 404
- Error Message: The user login profile does not exist.
- 解决方案: 未给予用户启用控制台登录, 请启用子用户控制台登录。

EntityNotExist.User.MFADevice

- HTTP Status: 404
- Error Message: The user has not bound any MFA device.
- 解决方案: 请给予用户绑定 MFA 设备。

EntityNotExist.User.Policy

- HTTP Status: 404
- Error Message: The indicate policy of the user does not exist.
- 解决方案: 子用户没有这个策略, 请确认授权策略名称和子用户名是否正确。

EntityNotExist.User.Group

- HTTP Status: 404
- Error Message: The user has not joined the group.
- 解决方案: 群组没有这个子用户, 请确认群组名称和子用户名是否正确。

EntityNotExist.VirtualMFADevice

- HTTP Status: 404
- Error Message: The virtual mfa device does not exist.
- 解决方案: 虚拟 MFA 设备不存在, 请确认虚拟 MFA 设备名称是否正确。

EntityNotExist.Group

- HTTP Status: 404
- Error Message: The group does not exist.

- 解决方案：请确认群组是否真实存在。

EntityNotExist.Group.Policy

- HTTP Status: 404
- Error Message: The indicate policy attached to the group does not exist.
- 解决方案：请确认群组是否有这个策略，并确认群组名、授权策略名是否正确。

EntityNotExist.Policy

- HTTP Status: 404
- Error Message: The policy does not exist.
- 解决方案：请确认策略是否真实存在。

EntityNotExist.Policy.Version

- HTTP Status: 404
- Error Message: The policy version does not exist.
- 解决方案：请确认策略版本是否真实存在或者已被删除。

HTTP Status 409

EntityAlreadyExistsEntityAlreadyExists.User

- HTTP Status: 409
- Error Message: The user does already EXIST.
- 解决方案：有相同用户名的 RAM 子用户已经存在，不可重新创建。

EntityAlreadyExists.User.LoginProfile

- HTTP Status: 409
- Error Message: The user login profile does already EXIST.
- 解决方案：已经启用子用户控制台登录，不需要再次启用。

EntityAlreadyExists.User.MFADevice

- HTTP Status: 409
- Error Message: The user has already bound a MFA device.
- 解决方案：子用户已经被绑定过 MFA 设，不需要再次绑定。

EntityAlreadyExists.User.Group

- HTTP Status: 409
- Error Message: The user has already joined the group.
- 解决方案：子用户已经存在于群组中，不需要再次加入群组。

EntityAlreadyExists.User.Policy

- HTTP Status: 409
- Error Message: The user has already been attached this policy.
- 解决方案: 子用户已经有此策略, 不需要再次授权此策略给子用户。

EntityAlreadyExists.VirtualMFADevice

- HTTP Status: 409
- Error Message: The virtual mfa device does already EXIST.
- 解决方案: 虚拟 MFA 设备已经存在, 不需要再次创建虚拟 MFA 设备。

EntityAlreadyExists.VirtualMFADevice.User

- HTTP Status: 409
- Error Message: The virtual mfa device does already been bound to a user.
- 解决方案: 虚拟 MFA 设备已经绑定给子用户, 不需要再次绑定虚拟 MFA 设备给子用户。

EntityAlreadyExists.Group

- HTTP Status: 409
- Error Message: The group does already EXIST.
- 解决方案: 有相同群组名的群组已经存在, 不可重复创建。

EntityAlreadyExists.Group.Policy

- HTTP Status: 409
- Error Message: The group has already been attached this policy.
- 解决方案: 群组已经有此策略, 不可重复授权相同的策略给群组。

EntityAlreadyExists.Policy

- HTTP Status: 409
- Error Message: The group does already EXIST.
- 解决方案: 策略已经存在, 不可重复创建同样策略名字且同样策略类型的授权策略。

EntityAlreadyExists.AccountAlias

- HTTP Status: 409
- Error Message: The account alias does already exist.
- 解决方案: 账户别名已经存在, 不可再设置。可使用 ClearAccountAlias 清除账户别名后重新设置。

DeleteConflictDeleteConflict.User.Group

- HTTP Status: 409
- Error Message: The user CAN NOT be in any group while deleting the user.
- 解决方案: 删除 RAM 子用户时, 子用户不能在群组中。请将子用户从群组中移除, 然后再删除 RAM 子用户。

DeleteConflict.User.AccessKey

- HTTP Status: 409
- Error Message: The user CAN NOT has any access key while deleting the user.
- 解决方案: 删除子用户时, 子用户不能有 AccessKey。请先删除子用户的 AccessKey, 然后再删除子用户。

DeleteConflict.User.LoginProfile

- HTTP Status: 409
- Error Message: The user CAN NOT has any login profile while deleting the user.
- 解决方案: 删除子用户时, 子用户不能有 login profile。请先删除子用户的 login profile, 然后再删除子用户。

DeleteConflict.User.MFADevice

- HTTP Status: 409
- Error Message: The user CAN NOT has any mfa device while deleting the user.
- 解决方案: 删除子用户时, 子用户不能有 MFA 设备。请先从子用户解绑 MFA 设备, 然后再删除子用户。

DeleteConflict.User.Policy

- HTTP Status: 409
- Error Message: The user CAN NOT has any attached policy while deleting the user.
- 解决方案: 删除子用户时, 子用户不能有授权策略。请先给子用户取消授权, 然后再删除。

DeleteConflict.VirtualMFADevice.User

- HTTP Status: 409
- Error Message: The virtual mfa device CAN NOT been bound to any user while deleting it.
- 解决方案: 删除虚拟 MFA 设备时, 虚拟 MFA 设备不能被绑定子用户。请先将虚拟 MFA 设备从子用户解绑, 然后再删除虚拟MFA设备。

DeleteConflict.Group.User

- HTTP Status: 409

- **Error Message:** The group CAN NOT has any user member while deleting the group.
- **解决方案:** 删除群组时, 群组中不能存在子用户。请将子用户从群组中移除, 然后再删除群组。

DeleteConflict.Group.Policy

- **HTTP Status:** 409
- **Error Message:** The entity CAN NOT has any attached policy while deleting the group.
- **解决方案:** 删除群组时, 群组不能有授权策略。请取消对群组的授权, 然后再删除群组。

DeleteConflict.Policy.User

- **HTTP Status:** 409
- **Error Message:** The policy CAN NOT been attached to any user while deleting the policy.
- **解决方案:** 删除策略时, 策略不能已经被授权给子用户。请将此策略从子用户解除授权, 然后再删除策略。

DeleteConflict.Policy.Group

- **HTTP Status:** 409
- **Error Message:** The policy CAN NOT been attached to any group while deleting the policy.
- **解决方案:** 删除策略时, 策略不能已经被授权给群组。请将策略从群组解除授权, 然后再删除策略。

DeleteConflict.Policy.Version

- **HTTP Status:** 409
- **Error Message:** The policy CAN NOT has any version except the default version.
- **解决方案:** 当前策略有超过 1 个版本, 不允许删除。请删除其它版本, 当策略只有默认版本时才可以被删除。

DeleteConflict.Policy.Version.Default

- **HTTP Status:** 409
- **Error Message:** The default policy version CAN NOT been deleted directly.
- **解决方案:** 当前策略版本为默认版本, 不能删除。请确认要删除的版本号。

DeleteConflict.Role.Policy

- **HTTP Status:** 409

- **Error Message:** The role CAN NOT has any attached policy while deleting the role.
- **解决方案:** 删除角色时, 角色不能有授权策略。请取消对角色的授权, 然后再删除角色。

LimitExceededLimitExceeded.User

- **HTTP Status:** 409
- **Error Message:** The count of users beyond the current limits.
- **解决方案:** 子用户数量超过当前上限 (子用户数上限 100)。当前子用户数据已经到达上限, 不可再创建新的RAM子用户。

LimitExceeded.User.AccessKey

- **HTTP Status:** 409
- **Error Message:** The access key count of the user access keys beyond the current limits.
- **解决方案:** 子用户的 AccessKey 数量超过上限 (子用户 AccessKey 上限 2 个)。

LimitExceeded.User.Group

- **HTTP Status:** 409
- **Error Message:** The count of groups the target user joined beyond the current limits.
- **解决方案:** 子用户加入的群组数量超过上限 (子用户可加入的群组上限为 5)。

LimitExceeded.User.Policy

- **HTTP Status:** 409
- **Error Message:** The policy count of the user attached policies beyond the current limits.
- **解决方案:** 子用户的策略数超过上限 (单个子用户的系统策略数上限为 20, 单个子用户的自定义策略数上限为 5)。

LimitExceeded.VirtualMFADevice

- **HTTP Status:** 409
- **Error Message:** The count of virtual mfa devices beyond the current limits.
- **解决方案:** 虚拟 MFA 设备数超过上限 (虚拟 MFA 设备数上限 100)。

LimitExceeded.Group

- **HTTP Status:** 409
- **Error Message:** The count of groups beyond the current limits.
- **解决方案:** 群组数量超过上限 (群组数量上限 50)。

LimitExceeded.Group.Policy

- HTTP Status: 409
- Error Message: The policy count of the group beyond the current limits.
- 解决方案: 群组的授权策略数量超过上限 (单个群组的系统策略数量上限 20, 单个群组的自定义策略数量上限 20)。

LimitExceeded.Policy

- HTTP Status: 409
- Error Message: The count of policy beyond the current limits.
- 解决方案: 自定义策略数量超过上限 (自定义策略数量上限 200)。

LimitExceeded.Policy.Version

- HTTP Status: 409
- Error Message: The count of policy version beyond the current limits.
- 解决方案: 现存自定义策略版本超过上限 (现存自定义策略版本上限 5)。

LimitExceeded.Role

- HTTP Status: 409
- Error Message: The count of roles beyond the current limits.
- 解决方案: 角色数量超过上限 (角色数量上限 100)。

LimitExceeded.Role.Policy

- HTTP Status: 409
- Error Message: The count of policies attached to the role beyond the current limits
-
- 解决方案: 角色的策略数超过上限 (单个角色的系统策略数上限 20, 单个角色的自定义策略数上限 5)。

HTTP Status 500

InternalError

- HTTP Status: 500
- Error Message: An internal error occurred.
- 解决方案: 服务内部错误。可通过工单联系售后技术工程师。

9.3 数据类型

9.3.1 User

用户信息。

名称	类型	示例值	描述
UserId	String	1227489245 380721	用户唯一标识。
UserName	String	zhangqiang	用户名称。
DisplayName	String	张强	显示名称。
Email	String	zhangqiang@ example.com	RAM用户的邮箱。
MobilePhone	String	86-18600008888	RAM用户手机号。
LastLoginDate	String	2015-01-23T12:33 :18Z	上次使用密码登录时间。
Comments	String	这是一位云计算工 程师	备注。
CreateDate	String	2015-01-23T12:33 :18Z	创建时间。
UpdateDate	String	2015-02-11T03:15 :21Z	更新时间。

9.3.2 LoginProfile

用户登录配置。

名称	类型	示例值	描述
UserName	String	zhangqiang	用户名。
CreateDate	String	2015-01-23T12:33 :18Z	创建时间。
MFABindRequired	Boolean	false	要求必须绑定多因素认证设备。
PasswordResetRequired	Boolean	false	要求下次登录时重设密码。

9.3.3 MFADevice

多因素认证设备。

名称	类型	示例值	描述
SerialNumber	String	acs:ram:: 1234567890123: mfa/device001	设备序列号。

9.3.4 VirtualMFADevice

虚拟多因素认证设备。

名称	类型	示例值	描述
User			绑定用户的基本信息。
└─UserName	String	zhangqiang	用户名称。
└─UserId	String	28877424437521 ****	用户唯一标识。
└─DisplayName	String	张强	显示名称。
Base32StringSeed	String	DSF98HAD98 2KJA9SDFNA S9D8FU839B 8ADHBGSD7A8	多因素认证设备密钥。
QRCodePNG	String	YXNkZmFzZD lmeW5hc2Q5 OGZoODd4bX JmcThhaGU5 aSBmYXNkZi BzYWWRmIGFG IDRxd2VjIG EgdHEzdCAG	密钥二维码PNG，使用Base64编码。
SerialNumber	String	acs:ram:: 1234567890123: mfa/device001	设备序列号。
ActivateDate	String	2015-02-18T17:22 :08Z	激活日期。

9.3.5 AccessKey

访问密钥。

名称	类型	示例值	描述
AccessKeyId	String	0wNEpMMLzy7szvai	访问密钥标识。
AccessKeySecret	String	PupkTg8jdm au1cXxYacg E736Pj4cA	访问密钥。
CreateDate	String	2015-01-23T12:33:18Z	创建时间。
Status	String	Active	状态。取值为激活或禁用。

9.3.6 Group

组信息类型。

名称	类型	示例值	描述
GroupName	String	Dev-Team	组名称。
CreateDate	String	2015-01-23T12:33:18Z	创建日期。
UpdateDate	String	2015-02-11T03:15:21Z	修改日期。
JoinDate	String	2015-01-23T12:33:18Z	加入日期。
Comments	String	开发团队	备注信息。

9.3.7 Role

角色。

名称	类型	示例值	描述
RoleName	String	ECSAdmin	角色名称。
RoleId	String	901234567890123	角色ID。
Arn	String	acs:ram:: 1234567890 123456:role/ ECSAdmin	角色的资源描述符。

名称	类型	示例值	描述
AssumeRolePolicyDocument	String	{ "Statement": [{ "Action": "sts:AssumeRole", "Effect": "Allow", "Principal": { "RAM": "acs:ram::123456789012345678:root" } }], "Version": "1" }	角色扮演授权策略。
CreateDate	String	2015-01-23T12:33:18Z	创建时间。
UpdateDate	String	2015-01-23T12:33:18Z	更新时间。
Description	String	ECS管理角色	角色的文字描述。

9.3.8 Policy

权限策略。

名称	类型	示例值	描述
PolicyName	String	OSS-Administrator	授权策略名称。
PolicyType	String	Custom	授权策略类型。
DefaultVersion	String	v1	默认版本。
AttachmentCount	Integer	3	引用次数。
CreateDate	String	2015-01-23T12:33:18Z	创建时间。
UpdateDate	String	2015-01-23T12:33:18Z	修改时间。
Description	String	OSS管理员权限	授权策略描述。

9.3.9 PolicyVersion

权限策略版本。

名称	类型	示例值	描述
VersionId	String	v3	权限策略版本标识。
PolicyDocument	String	{ "Statement": [{ "Action": ["oss:*"], "Effect": "Allow", "Resource": ["acs:oss:*:*:*"] }, { "Version": "1" }] }	权限策略内容。
IsDefaultVersion	Boolean	false	是否默认版本。
CreateDate	String	2015-01-23T12:33:18Z	创建时间。

9.3.10 PasswordPolicy

密码策略。

名称	类型	示例值	描述
HardExpiry	Boolean	false	过期后是否可以登录。默认为false，过期后可登录。
MaxLoginAttempts	Integer	5	一小时内使用错误密码尝试登录最大次数（重置密码可清除尝试登录计数）。
MaxPasswordAge	Integer	0	密码有效期，单位是天（重置密码将重置密码过期时间）。0表示不启用密码过期策略，默认不启用。
MinimumPasswordLength	Integer	12	最小密码长度。
PasswordReusePrevention	Integer	0	禁止使用前N次密码。0表示不启用历史密码检查策略，默认不启用。
RequireLowercaseCharacters	Boolean	true	必须包含小写字母。
RequireNumbers	Boolean	true	必须包含数字。
RequireSymbols	Boolean	true	必须包含字符。

名称	类型	示例值	描述
RequireUppercaseCharacters	Boolean	true	必须包含大写字母。

9.3.11 SecurityPreference

安全首选项。

返回参数

名称	类型	示例值	描述
AccessKeyPreference			访问密钥首选项。
└ AllowUserToManageAccessKeys	Boolean	false	是否允许用户管理自己的访问密钥。
LoginProfilePreference			登录首选项。
└ AllowUserToChangePassword	Boolean	true	是否允许用户修改自己的密码。
└ EnableSaveMFATicket	Boolean	true	是否允许在用户在登录时保存MFA验证票据。
└ LoginNetworkMasks	String	10.0.0.0/8	登录掩码，默认空字符串，不限制登录IP。
└ LoginSessionDuration	Integer	6	用户登录有效期，单位：小时。
MFAPreference			多因素认证首选项。
└ AllowUserToManageMFADevices	Boolean	true	是否允许用户管理自己绑定或解绑多因素认证设备。
PublicKeyPreference			公钥首选项。

名称	类型	示例值	描述
└ AllowUserToManagePublicKeys	Boolean	false	是否允许用户管理自己的公钥。

9.4 关于RAM限制

请参考RAM API文档的[#unique_152](#)。