

目录

目录	1
请求结构	11
请求结构	11
公共参数	11
返回结果	12
调用成功	12
调用失败	12
签名机制	12
签名过程	12
1. 构造规范化请求字符串	12
2. 计算签名。	13
3. 将签名值作为Signature参数值添加到请求参数中。	13
签名过程示例	13
签名DEMO	14
php	14
java	14
golang	16
python3	17
新建子用户	17
请求参数	17
错误码（调用不成功）	17
示例（调用成功）	18
删除子用户	18
限制说明	18
请求参数	18
错误码（调用不成功）	19
示例	19
请求示例	19
返回示例	19
查询子用户基本信息	19
查询子用户基本信息（GetUser）	19
请求参数	19
返回元素（调用成功）	19
错误码（调用不成功）	19
示例	19
请求示例	19
返回示例	20
更新子用户基本信息	20
请求参数	20
返回元素（调用成功）	20
错误码（调用不成功）	20
示例（调用成功）	21
请求示例	21
返回示例	21
查询子用户列表	21
请求参数	21
返回元素（调用成功）	22
错误码（调用不成功）	22
示例（调用成功）	22
请求示例	22

返回示例	22
新建子用户登录配置	22
接口描述	23
请求参数	23
返回元素（调用成功）	23
错误码（调用不成功）	23
示例（调用成功）	23
请求示例	23
返回示例	23
删除子用户登录配置	24
请求参数	24
返回元素（调用成功）	24
错误码（调用不成功）	24
示例（调用成功）	24
请求示例	24
返回示例	24
查询子用户登录配置	24
请求参数	24
返回元素（调用成功）	25
错误码（调用不成功）	25
示例（调用成功）	25
请求示例	25
返回示例	25
更新子用户登录配置	25
请求参数	25
返回元素（调用成功）	26
错误码（调用不成功）	26
示例（调用成功）	26
请求示例	26
返回示例	26
修改子用户密码	26
请求参数	26
返回元素（调用成功）	27
错误码（调用不成功）	27
示例（调用成功）	27
请求示例	27
返回示例	27
获取子用户登录态	27
请求参数	27
返回元素（调用成功）	27
错误码（调用不成功）	28
示例	28
请求示例	28
返回示例	28
创建成员账号	28
请求参数	28
返回元素（调用成功）	28
错误码（调用不成功）	28
示例	29
请求示例	29
返回示例	29
获取子用户AK最后使用时间	29

请求参数	29
返回元素（调用成功）	29
示例（调用成功）	29
请求示例	29
返回示例	29
生成访问密钥	30
请求参数	30
返回元素（调用成功）	30
错误码（调用不成功）	30
示例（调用成功）	30
请求示例	30
返回示例	30
删除访问密钥	31
请求参数	31
返回元素（调用成功）	31
错误码（调用不成功）	31
示例（调用成功）	31
请求示例	31
返回示例	31
查询访问密钥列表	32
请求参数	32
返回元素（调用成功）	32
错误码（调用不成功）	32
示例（调用成功）	32
请求示例	32
返回示例	32
更新访问密钥	33
请求参数	33
返回元素（调用成功）	33
错误码（调用不成功）	33
示例（调用成功）	33
请求示例	33
返回示例	33
新建策略	34
请求参数	34
返回元素（调用成功）	34
错误码（调用不成功）	34
示例（调用成功）	35
请求示例	35
返回示例	35
删除策略	35
请求参数	35
返回元素（调用成功）	36
错误码（调用不成功）	36
示例（调用成功）	36
请求示例	36
返回示例	36
查询策略元数据信息	36
请求参数	36
返回元素（调用成功）	36
错误码（调用不成功）	37
示例（调用成功）	37

请求示例	37
返回示例	37
查询策略信息列表	37
请求参数	37
返回元素（调用成功）	38
错误码（调用不成功）	38
示例（调用成功）	38
请求示例	38
返回示例	38
新建策略版本内容	39
请求参数	39
返回元素（调用成功）	39
错误码（调用不成功）	39
示例（调用成功）	40
请求示例	40
返回示例	40
删除策略版本内容	40
请求参数	40
返回元素（调用成功）	41
错误码（调用不成功）	41
示例（调用成功）	41
请求示例	41
返回示例	41
查询策略版本内容	41
请求参数	41
返回元素（调用成功）	42
错误码（调用不成功）	42
示例（调用成功）	42
请求示例	42
返回示例	42
查询策略版本列表	42
请求参数	42
返回元素（调用成功）	43
错误码（调用不成功）	43
示例（调用成功）	43
请求示例	43
返回示例	43
设定默认策略版本	44
请求参数	44
返回元素（调用成功）	44
错误码（调用不成功）	44
示例（调用成功）	44
请求示例	44
返回示例	44
附加用户策略	44
请求参数	45
返回元素（调用成功）	45
错误码（调用不成功）	45
示例（调用成功）	45
请求示例	45
返回示例	45
分离用户策略	46

请求参数	46
返回元素 (调用成功)	46
错误码 (调用不成功)	46
示例 (调用成功)	46
请求示例	46
返回示例	46
查询用户附加策略列表	46
请求参数	46
返回元素 (调用成功)	47
错误码 (调用不成功)	47
示例 (调用成功)	47
请求示例	47
返回示例	47
查询策略附加实体列表	48
请求参数	48
返回元素 (调用成功)	48
错误码 (调用不成功)	48
示例 (调用成功)	48
请求示例	48
返回示例	48
添加用户到用户组	49
请求参数	49
返回元素	49
错误码	49
示例	49
请求示例	49
返回示例	49
附加策略到用户组	50
请求参数	50
返回元素	50
错误码	50
示例	50
请求示例	50
返回示例	50
创建用户组	50
创建用户组(CreateGroup)	50
描述	50
请求参数	50
返回元素	51
错误码	51
示例	51
请求示例	51
返回示例	51
删除用户组	51
请求参数	51
错误码	51
示例	52
请求示例	52
返回示例	52
从用户组分离策略	52
请求参数	52
返回元素	52

错误码	52
示例	52
请求示例	52
返回示例	52
获取用户组	53
请求参数	53
返回元素	53
错误码	53
示例	53
请求示例	53
返回示例	53
获取用户组权限列表	54
获取用户组权限列表(ListGroupPolicies)	54
请求参数	54
返回元素	54
错误码	54
示例	54
请求示例	54
返回示例	54
获取用户组列表	54
请求参数	54
返回元素	55
错误码	55
示例	55
请求示例	55
返回示例	55
获取用户所属用户组列表	55
请求参数	55
返回元素	55
错误码	55
示例	56
请求示例	56
返回示例	56
从用户组删除用户	56
从用户组删除用户(RemoveUserFromGroup)	56
描述	56
请求参数	56
返回元素	56
错误码	56
示例	57
请求示例	57
返回示例	57
更新用户组	57
请求参数	57
返回元素	57
错误码	57
示例	57
请求示例	57
返回示例	57
创建角色	58
概述	58
请求参数	58

返回元素（调用成功）	58
错误码（调用不成功）	58
示例（调用成功）	58
请求示例	58
返回示例	58
删除角色	59
请求参数	59
返回元素（调用成功）	59
错误码（调用不成功）	59
示例（调用成功）	59
请求示例	59
返回示例	59
查询账号的角色列表	59
请求参数	59
返回元素（调用成功）	60
错误码（调用不成功）	60
示例（调用成功）	60
请求示例	60
返回示例	60
查询角色基本信息	60
请求参数	60
返回元素（调用成功）	60
错误码（调用不成功）	61
示例（调用成功）	61
请求示例	61
返回示例	61
更新角色的信任策略	61
概述	61
openAPI接口	61
请求参数	61
返回元素（调用成功）	61
错误码（调用不成功）	61
示例（调用成功）	62
请求示例	62
返回示例	62
附加角色访问策略	62
概述	62
请求参数	62
返回元素（调用成功）	62
错误码（调用不成功）	62
示例（调用成功）	62
请求示例	63
返回示例	63
分离角色访问策略	63
请求参数	63
返回元素（调用成功）	63
错误码（调用不成功）	63
示例（调用成功）	63
请求示例	63
返回示例	63
查询角色附加的策略列表	63
概述	63

请求参数	64
返回元素（调用成功）	64
错误码（调用不成功）	64
示例（调用成功）	64
请求示例	64
返回示例	64
获取角色的临时身份	65
概述	65
请求参数	65
返回元素（调用成功）	65
错误码（调用不成功）	65
示例（调用成功）	65
请求示例	65
返回示例	65
创建虚拟设备	66
请求参数	66
返回元素（调用成功）	66
错误码（调用不成功） {#错误码（调用不成功）}	66
示例（调用成功）	66
请求示例	66
返回示例	66
查询虚拟设备列表	67
请求参数	67
错误码（调用不成功）	67
示例（调用成功）	67
请求示例	67
返回示例	67
激活虚拟设备	68
请求参数	68
错误码（调用不成功） {#错误码（调用不成功）}	68
示例（调用成功）	68
请求示例	68
返回示例	68
解绑虚拟设备	69
请求参数	69
错误码（调用不成功） {#错误码（调用不成功）}	69
示例（调用成功）	69
请求示例	69
返回示例	69
删除虚拟设备	69
请求参数	69
错误码（调用不成功） {#错误码（调用不成功）}	69
示例（调用成功）	70
请求示例	70
返回示例	70
获取虚拟设备	70
请求参数	70
错误码（调用不成功）	70
示例（调用成功）	70
请求示例	70
返回示例	70
创建项目	71

请求参数	71
请求示例	71
获取主账号/子用户下所有项目列表	71
请求参数	71
请求示例	71
查询项目成员	72
请求参数	72
示例	72
请求示例	72
返回示例	72
移除项目成员	73
请求参数	73
错误码（调用不成功）	73
示例	73
请求示例	73
返回示例	73
获取项目里面资源列表	73
请求参数	73
请求示例	74
批量查询资源所属的项目ID列表	74
请求参数	74
请求示例	74
查询账户概要信息	75
请求参数	75
返回元素（调用成功）	75
错误码（调用不成功）	75
示例（调用成功）	75
请求示例	75
返回示例	75
User	77
描述	77
内容	77
Role	77
描述	77
内容	77
LoginProfile	77
描述	77
内容	78
AccessKey	78
描述	78
内容	78
AccessKeyMetadata	78
描述	78
内容	78
Policy	79
描述	79
内容	79
PolicyVersion	79
描述	79
内容	79
AttachedPolicy	80
描述	80

内容	80
PolicyUser	80
描述	80
内容	80
SummaryMap	80
描述	80
内容	80
PasswordPolicy	81
描述	81
内容	81
VirtualMFADevice	81
描述	81
内容 {	81
AssumeRoleResult	82
AssumeRoleResult {#73-assumeroleresult}	82
描述	82
AssumedRoleUser 内容	82
Credentials 内容	82
其他内容	82
UserSession	82
描述	83
内容	83
KRN查询列表	83

请求结构

请求结构

客户调用金山云访问控制服务的openAPI接口是通过向指定服务地址发送请求，并按照openAPI文档说明在请求中添加相应的公共参数和接口参数来完成的。

访问控制openAPI的请求结构组成如下：

1. 服务地址

访问控制的服务接入地址为：`iam.api.ksyun.com`

2. 通信协议

支持通过 HTTP 或 HTTPS 两种方式进行请求通信，推荐使用安全性更高的 HTTPS方式发送请求。

3. 请求方法

访问控制的openAPI同时支持GET和POST请求，推荐使用GET请求方式。

4. 请求参数

金山云openAPI请求包含两类参数：**公共请求参数**和**业务请求参数**。其中，公共请求参数是每个接口都要用到的请求参数，具体可参见[公共参数](#)小节；业务请求参数是各个接口所特有的，具体见各个接口的“请求参数”描述。

5. 字符编码

请求及返回结果都使用UTF-8字符集进行编码。

公共参数

公共参数是每个openAPI接口都需要使用的请求参数。支持GET和POST两种HTTP方法。

- GET请求：放在url的query。面。例如：`iam.api.ksyun.com?{业务参数}&{公共参数}`。
- POST请求：放在http body里面。例如：`{业务参数}&{公共参数}`。

名称	类型	是否必须参数	描述
Accesskey	String	是	用户在控制台创建的Accesskey
Service	String	是	服务名称，提供open-api的服务英文简称，访问控制服务固定值iam
Action	String	是	操作接口名，与调用的具体openAPI相关
Version	String	是	接口版本号，访问控制服务接口当前只支持一个版本，即2015-11-01
Timestamp	String	是	时间，UTC格式，例如：2019-08-13T17:18:36Z
SignatureVersion	String	是	签名版本号，固定值：1.0
SignatureMethod	String	是	签名算法，固定值：HMAC-SHA256
Signature	String	是	签名，具体请查看 签名机制
Region	String	否	区域，默认cn-beijing-6。不同服务支持的region不同，访问控制服务使用的region为cn-beijing-6
SecurityToken	String	否	安全令牌，在使用临时ak(STS 角色扮演获取的ak/sk)，需要传该字段，如果使用GET方法，需要对该字段进行urlencode
DryRun	Boolean	否	检查当前调用者是否有权限执行相关操作，而不是真的调用执行相关操作
Format	String	否	指定响应格式，固定值：json

示例

```
https://iam.api.ksyun.com/?
Action=ListUsers
&Accesskey=AKLTAGFB7-e_Qtest
&Service=iam
&Version=2015-11-01
&Timestamp=2020-08-05T07:16:22Z
&SignatureVersion=1.0
&SignatureMethod=HMAC-SHA256
&Signature=6f00100528da6630b786a9326813916c4b6e77dgagda2d4e4a1b05c7e
```

&业务参数

返回结果

调用金山云的openAPI服务，调用成功，返回的HTTP状态码（Status）为200；调用失败，返回4xx 或5xx的HTTP状态码（Status）。

金山云的访问控制服务的调用返回的数据格式支持xml和json两种，默认返回xml格式，可通过设置HTTP Header Accept=application/json来改变返回数据格式。

调用成功

- xml格式示例

```
<!--结果的根结点-->
<接口名称+Response>
  <ResponseMetadata>
    <RequestId>4C467B38-3910-447D-87BC-AC049166F216</RequestId>
  </ResponseMetadata>
  <!--返回结果数据-->
</接口名称+Response>
```

- json格式示例

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

调用失败

调用接口失败，不会返回结果数据；HTTP请求返回一个4xx或5xx的HTTP状态码，返回的HTTP消息体中包含具体的错误代码（code）及错误信息（message）；与调用成功一样还包含请求ID（RequestId）。

- json格式示例

```
{
  "RequestId": "68093a99-2f63-4f39-8f70-3047ab8ecb5b",
  "Error": {
    "Type": "Sender",
    "Code": "InvalidParameterValue",
    "Message": "An invalid or out-of-range value was supplied for the input parameter PathPrefix."
  }
}
```

- 当请求出错时建议可以先根据该错误码在[公共错误码](#)（所有业务都可能出现的错误码为公共错误码）和当前接口对应的错误码列表里面查找对应原因和解决方案。
- 如果您找不到错误原因时，可以联系金山云客服，并提供RequestId，以便我们尽快帮您解决问题。

签名机制

签名过程

1. 构造规范化请求字符串

第一步：请求参数排序

- 请求参数包括公共参数和业务参数，不包含公共参数Signature
- 排序规则以参数名按照字典排序。

第二步：请求参数编码 使用UTF-8字符集按照[RFC3986](#)规则编码请求参数和参数取值，编码规则如下：

- 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（大写）。

不同语言实现的URLEncode方式不同，为了得到[RFC3986](#)规则的编码，分别说明如下：

- 如果您使用的是Java中的java.net.URLEncoder，可以先用URLEncoder.encode方法编码，随后将编码后的字符中加号（+）替换为%20、星号（*）替换为%2A、%7E替换为波浪号（~）即可；
- 如果您使用的是golang中的net/url，可以用url.Values.Encode方法编码，随后将编码后的字符中加号（+）替换为%20即可；

- 如果您使用的是php中的rawurlencode方法，不需要替换；
- 如果您使用的是python3中的urllib.request，可以用urllib.request.quote方法，需要指定字符串中的波浪号(~)不编码，例如urllib.request.quote(str, '~')，str为待编码字符串；

第三步：请求参数拼接成CanonicalizedQueryString 每对URLEncode后的参数名称和参数值，用=进行连接。每对之间使用&进行连接。得到规范化请求字符串CanonicalizedQueryString。

2. 计算签名。

```
sign = hash_hmac('sha256', CanonicalizedQueryString, sk)
```

sign值为签名算法返回的16进制格式小写字符串

签名样例：

5346bfefeb3f2162e2459e09a52b640584e5f1aa5012b15c6b85388680d4663e 计算签名时使用的sk为Accesskey对应的密钥，使用的哈希算法是：HMAC-SHA256。

3. 将签名值作为Signature参数值添加到请求参数中。

签名过程示例

该样例以iam服务的CreateUser为例，点击[查看该API文档](#)。

1、构造规范化请求字符串：CanonicalizedQueryString

//请求参数数组：

```
array (
  'Accesskey' => 'AKLTXQVF0p0mS6aahIrD5r0B3Q', //公共参数
  'Service' => 'iam', //公共参数
  'Action' => 'CreateUser', //公共参数
  'Version' => '2015-11-01', //公共参数
  'Timestamp' => '2021-08-12T02:47:36Z', //公共参数
  'SignatureVersion' => '1.0', //公共参数
  'SignatureMethod' => 'HMAC-SHA256', //公共参数
  'UserName' => 'Ttest', //业务参数
  'RealName' => '周四测试', //业务参数
  'Email' => 'zsce@kingsoft.com', //业务参数
  'Remark' => '~ce shi*#|+', //业务参数
)
```

第一步：请求参数排序

//排序结果如下：

```
array (
  'Accesskey' => 'AKLTXQVF0p0mS6aahIrD5r0B3Q',
  'Action' => 'CreateUser',
  'Email' => 'zsce@kingsoft.com',
  'RealName' => '周四测试',
  'Remark' => '~ce shi*#|+',
  'Service' => 'iam',
  'SignatureMethod' => 'HMAC-SHA256',
  'SignatureVersion' => '1.0',
  'Timestamp' => '2021-08-12T02:47:36Z',
  'UserName' => 'Ttest',
  'Version' => '2015-11-01',
)
```

第二步：请求参数编码

//编码结果如下：

```
array (
  'Accesskey' => 'AKLTXQVF0p0mS6aahIrD5r0B3Q',
  'Action' => 'CreateUser',
  'Email' => 'zsce%40kingsoft.com',
  'RealName' => '%E5%91%A8%E5%9B%9B%E6%B5%8B%E8%AF%95',
  'Remark' => '~ce%20shi%2A%25%23%7C%2B',
  'Service' => 'iam',
  'SignatureMethod' => 'HMAC-SHA256',
  'SignatureVersion' => '1.0',
  'Timestamp' => '2021-08-12T02%3A47%3A36Z',
)
```

```
'UserName' => 'Ttest',
'Version' => '2015-11-01',
)
```

第三步：请求参数拼接成CanonicalizedQueryString

//拼接结果如下

```
Accesskey=AKLTXXQVF0p0mS6aahIrD5r0B3Q&Action=CreateUser&Email=zsce%40kkingsoft.com&RealName=%E5%91%A8%E5%9B%9B%E6%B5%8B%E8%AF%95&Remark=~ce%20shi%2A%25%23%7C%2B&Service=iam&SignatureMethod=HMAC-SHA256&SignatureVersion=1.0&Timestamp=2021-08-12T02:3A47%3A36Z&UserName=Ttest&Version=2015-11-01
```

2、计算签名

```
sk = '0MouU5PTLh6y9E9Ioe3K411jt99VqyQSBXgAcDY1o49R31vUIzb6e/efZCFDmtFlzw=='
CanonicalizedQueryString = "Accesskey=AKLTXXQVF0p0mS6aahIrD5r0B3Q&Action=CreateUser&Email=zsce%40kkingsoft.com&RealName=%E5%91%A8%E5%9B%9B%E6%B5%8B%E8%AF%95&Remark=~ce%20shi%2A%25%23%7C%2B&Service=iam&SignatureMethod=HMAC-SHA256&SignatureVersion=1.0&Timestamp=2021-08-12T02:3A47%3A36Z&UserName=Ttest&Version=2015-11-01"
//sign = hash_hmac('sha256', CanonicalizedQueryString, sk)
sign: fc9088ab845949dac4040be9b7ce7859068b5c21d4c400fec8ee0cefb777f659
```

3、将签名值作为Signature参数值添加到请求参数中

//请求示例:

```
curl -s -L -X POST 'iam.api.ksyun.com' \
-H 'Accept: application/json' \
-H 'Content-Type: application/x-www-form-urlencoded' \
--data-urlencode 'Accesskey=AKLTXXQVF0p0mS6aahIrD5r0B3Q' \
--data-urlencode 'Service=iam' \
--data-urlencode 'Action=CreateUser' \
--data-urlencode 'Version=2015-11-01' \
--data-urlencode 'Timestamp=2021-08-12T02:47:36Z' \
--data-urlencode 'SignatureVersion=1.0' \
--data-urlencode 'SignatureMethod=HMAC-SHA256' \
--data-urlencode 'UserName=Ttest' \
--data-urlencode 'RealName=周四测试' \
--data-urlencode 'Email=zsce@kkingsoft.com' \
--data-urlencode 'Remark=~ce shi*%#|+' \
--data-urlencode 'Signature=fc9088ab845949dac4040be9b7ce7859068b5c21d4c400fec8ee0cefb777f659'
```

签名DEMO

php

//使用时把sign方法拷贝到代码工程内即可

```
function sign($params, $secret_key)
{
    ksort($params, SORT_STRING);

    $str_encode = '';
    foreach($params as $k=>$v) {
        $str_encode .= rawurlencode($k).'='.rawurlencode($v).'&';
    }
    $str_encode = substr($str_encode, 0, -1);

    return hash_hmac("sha256", $str_encode, $secret_key);
}
```

```
$arr =[
'Accesskey'=> 'AKLTXXQVF0p0mS6aahIrD5r0B3Q',
'Service'=> 'iam',
'Action'=> 'CreateUser',
'Version'=> '2015-11-01',
'Timestamp'=> '2021-08-12T02:47:36Z',
'SignatureVersion'=> '1.0',
'SignatureMethod'=> 'HMAC-SHA256',
'UserName'=> 'Ttest',
'RealName'=> '周四测试',
'Email'=> 'zsce@kkingsoft.com',
'Remark'=> '~ce shi*%#|+',
];
$sk = '0MouU5PTLh6y9E9Ioe3K411jt99VqyQSBXgAcDY1o49R31vUIzb6e/efZCFDmtFlzw==' ;
$signature = sign($arr, $sk);

var_dump($signature); //fc9088ab845949dac4040be9b7ce7859068b5c21d4c400fec8ee0cefb777f659
```

java

```
package com.ksyun.util;

import org.apache.commons.codec.digest.HmacAlgorithms;
import org.apache.commons.codec.digest.HmacUtils;

import java.io.UnsupportedEncodingException;
import java.net.URLEncoder;
import java.nio.charset.StandardCharsets;
import java.util.HashMap;
import java.util.Map;
import java.util.TreeMap;
import java.util.regex.Matcher;
import java.util.regex.Pattern;

/**
 * 使用时，把测试相关方法删掉。测试方法包括：main, test1, test2
 * 把这个类拷贝到自己的工程内
 * 签名调用：SignatureUtils.signature(params, secretKey)
 */
/**
 * @author chenxushao@kingsoft.com
 */
public class SignatureUtils {

    private static final Pattern ENCODED_CHARACTERS_PATTERN;

    static {
        StringBuilder pattern = new StringBuilder();
        pattern.append(Pattern.quote("+"))
            .append("|")
            .append(Pattern.quote("*"))
            .append("|")
            .append(Pattern.quote("%7E"))
            .append("|")
            .append(Pattern.quote("%2F"));
        ENCODED_CHARACTERS_PATTERN = Pattern.compile(pattern.toString());
    }

    /**
     * 注意空格( )会编码成+号，所以+最后需要替换成%20 (%20为空格)
     * 在URL编码后需对三种字符替换：加号(+) 替换成 %20、星号(*) 替换成 %2A、 %7E 替换成回波浪号(~)
     */
    private static String urlEncode(final String value, final boolean path) {
        if (value == null) {
            return "";
        }

        try {
            String encoded = URLEncoder.encode(value, StandardCharsets.UTF_8.name());
            Matcher matcher = ENCODED_CHARACTERS_PATTERN.matcher(encoded);
            StringBuffer buffer = new StringBuffer(encoded.length());

            while (matcher.find()) {
                String replacement = matcher.group(0);
                if ("+".equals(replacement)) {
                    replacement = "%20";
                } else if ("*".equals(replacement)) {
                    replacement = "%2A";
                } else if ("%7E".equals(replacement)) {
                    replacement = "~";
                } else if (path && "%2F".equals(replacement)) {
                    replacement = "/";
                }

                matcher.appendReplacement(buffer, replacement);
            }

            matcher.appendTail(buffer);
            return buffer.toString();
        } catch (UnsupportedEncodingException ex) {
            throw new RuntimeException(ex);
        }
    }

    private static String getCanonicalizedQueryString(Map<String, Object> params) {
        final Map<String, String> tmap = new TreeMap<>();
        for (Map.Entry<String, Object> entry : params.entrySet()) {
            String key = entry.getKey();
            Object value = entry.getValue();
            String encodedParamName = urlEncode(key, false);
            String encodedValues = urlEncode(value.toString(), false);
            tmap.put(encodedParamName, encodedValues);
        }
    }
}
```

```

    final StringBuilder sb = new StringBuilder();
    for (Map.Entry<String, String> entry : tmap.entrySet()) {
        if (sb.length() > 0) {
            sb.append("&");
        }
        sb.append(entry.getKey()).append('=').append(entry.getValue());
    }
    return sb.toString();
}

public static String signature(Map<String, Object> params, String secretKey) {
    String canonicalizedQueryString = getCanonicalizedQueryString(params);
    return new HmacUtils(HmacAlgorithms.HMAC_SHA_256, secretKey).hmacHex(canonicalizedQueryString);
}

public static void test2() {
    String accesskey = "AKLTXQVF0p0mS6aahIrD5r0B3Q";
    String secretKey = "0MovU5PTLh6y9E9Ioe3K411jt99VqyQSBXgAcDYlo49R3lvUzb6e/efZCFDmtFlzw=";

    Map<String, Object> params = new HashMap<>();
    params.put("Accesskey", accesskey); //公共参数
    params.put("Service", "iam"); //公共参数
    params.put("Action", "CreateUser"); //公共参数
    params.put("Version", "2015-11-01"); //公共参数
    params.put("Timestamp", "2021-08-12T02:47:36Z"); //公共参数
    params.put("SignatureVersion", "1.0"); //公共参数
    params.put("SignatureMethod", "HMAC-SHA256"); //公共参数
    params.put("UserName", "Ttest"); //api特有参数
    params.put("RealName", "周四测试"); //api特有参数
    params.put("Email", "zsce@kingsoft.com"); //api特有参数
    params.put("Remark", "~ce shi*%#|+"); //api特有参数

    System.out.println(signature(params, secretKey));
}

public static void test1() {
    String accesskey = "AKLTXQVF0p0mS6aahIrD5r0B3Q";
    String secretKey = "0MovU5PTLh6y9E9Ioe3K411jt99VqyQSBXgAcDYlo49R3lvUzb6e/efZCFDmtFlzw=";

    Map<String, Object> params = new HashMap<>();
    params.put("Accesskey", accesskey); //公共参数
    params.put("Service", "iam"); //公共参数
    params.put("Action", "GetUser"); //公共参数
    params.put("Version", "2015-11-01"); //公共参数
    params.put("Timestamp", "2021-08-06T07:45:36Z"); //公共参数
    params.put("SignatureVersion", "1.0"); //公共参数
    params.put("SignatureMethod", "HMAC-SHA256"); //公共参数
    params.put("UserName", "freestest"); //api特有参数

    System.out.println(signature(params, secretKey));
}

public static void main(String[] args) {
    test1(); //签名结果: 9294d873d0f921bed24b6089708b66fbdfc4a6ea0eb30ad21e73ce603b82fbb7
    test2(); //签名结果: fc9088ab845949dac4040be9b7ce7859068b5c21d4c400fec8ee0cef777f659
}
}

```

golang

```

package main

import (
    "crypto/hmac"
    "crypto/sha256"
    "encoding/hex"
    "fmt"
    "net/url"
    "strings"
)

//使用时把sign方法拷贝到代码工程内即可
func sign(params url.Values, sk string) string {
    strEncode := params.Encode()
    strEncode = strings.Replace(strEncode, "+", "%20", -1)
    h := hmac.New(sha256.New, []byte(sk))
    h.Write([]byte(strEncode))
    return hex.EncodeToString(h.Sum(nil))
}

func main() {
    params := url.Values{}
    params.Add("Accesskey", "AKLTXQVF0p0mS6aahIrD5r0B3Q")
    params.Add("Service", "iam")
}

```



```

params.Add("Action", "CreateUser")
params.Add("Version", "2015-11-01")
params.Add("Timestamp", "2021-08-12T02:47:36Z") //通过time.Now().UTC().Format("2006-01-02T15:04:05Z")实现
params.Add("SignatureVersion", "1.0")
params.Add("SignatureMethod", "HMAC-SHA256")
params.Add("UserName", "Ttest")
params.Add("RealName", "周四测试")
params.Add("Email", "zsce@kingsoft.com")
params.Add("Remark", "~ce shi*%#|+")

sk := "OMovU5PTLh6y9E9Ioe3K411jt99VqyQSBXgAcDYlo49R31vUIzb6e/efZCFDmtFlzw=="
signature := sign(params, sk)

fmt.Println(signature) // fc9088ab845949dac4040be9b7ce7859068b5c21d4c400fec8ee0cefb777f659
}

```

python3

```

import sys, os, base64, datetime, hashlib, hmac
import urllib.request

# 使用时把sign方法拷贝到代码工程内即可
def sign(params, secret_key):
    str_encode = ''
    param_keys = sorted(params.keys())
    for key in param_keys:
        str_encode += urllib.request.quote(key, '~') + '=' + urllib.request.quote(str(params[key]), '~') + '&'

    return hmac.new(bytes(secret_key, 'utf-8'), bytes(str_encode[:-1], 'utf-8'), hashlib.sha256).hexdigest()

arr = {
    'Accesskey' : 'AKLTXXQVF0p0mS6aahIrD5r0B3Q',
    'Service' : 'iam',
    'Action' : 'CreateUser',
    'Version' : '2015-11-01',
    'Timestamp' : '2021-08-12T02:47:36Z', # 使用如下的方式产生UTC格式的时间, datetime.datetime.utcnow()
    'SignatureVersion' : '1.0',
    'SignatureMethod' : 'HMAC-SHA256',
    'UserName' : 'Ttest',
    'RealName' : '周四测试',
    'Email' : 'zsce@kingsoft.com',
    'Remark' : '~ce shi*%#|+'
}

sk = 'OMovU5PTLh6y9E9Ioe3K411jt99VqyQSBXgAcDYlo49R31vUIzb6e/efZCFDmtFlzw=='
signature = sign(arr, sk)

print(signature) #结果: fc9088ab845949dac4040be9b7ce7859068b5c21d4c400fec8ee0cefb777f659

```

新建子用户

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是	确定值	[a-zA-Z]+	操作接口名, 系统规定参数, 取值: CreateUser
Path	String	否	最短1, 最长512	√ or √[:graph:]+√	可选参数, 用户名的路径, 如果未传入则默认值是/
UserName	String	是	最短1, 最长64	[\w+=, .@-]+	待创建的IAM用户的用户名
RealName	String	否	最短2, 最长128	[\u4e00-\u9fff]+	待创建的IAM用户的真实姓名
Phone	String	否	11位	`(13\ 14\ 15\ 17\ 18)\d{9}`	手机号
Email	String	否	最短2, 最长255	`([a-z0-9A-Z]+[-\ .])?+[a-z0-9A-Z]@[a-z0-9A-Z]+(-[a-z0-9A-Z])?.)+[a-zA-Z]{2,}`	邮箱
Remark	String	否	最短0, 最长128	中英文可见字符	备注

错误码 (调用不成功)

错误代码	描述	HTTP 状态码	语义
------	----	----------	----

MissingParameter	A required parameter for the specified action is not supplied.	400	忘记输入参数, 具体参数名在错误消息中说明 (UserName)
InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter.	400	参数格式/取值范围不对, 具体参数名在错误消息中说明 (UserName, Path, RealName)
EntityAlreadyExists	The request was rejected because it attempted to create a resource that already exists	409	资源已经存在, 具体实体名称在错误消息中说明 (UserName)
LimitExceeded	The request was rejected because it attempted to create resources beyond the current Ksyun account limits	409	资源数量超过限制, 具体限制在错误消息中说明 (User数量是100)

示例（调用成功）

请求示例

```
https://iam.ksyun.com/?Action=CreateUser&Path=/division_abc/subdivision_xyz/&UserName=Bob&RealName=金山云&公共请求参数
```

返回示例

- XML格式

```
<CreateUserResponse xmlns="https://iam.ksyun.com/doc/2015-11-01/">
  <CreateUserResult>
    <User>
      <UserId>待确定类型和位数</UserId>
      <Path>/division_abc/subdivision_xyz/</Path>
      <UserName>Bob</UserName>
      <Krn>krn:ksc:iam::12345678901234567890:user/division_abc/subdivision_xyz/Bob</Krn>
      <RealName>金山云</RealName>
      <CreateDate>2015-11-01T12:00:00Z</CreateDate>
      <Phone>13300001111</Phone>
      <Email>asdfsaf@sd.com</Email>
      <PhoneVerified>Yes</PhoneVerified>
      <EmailVerified>No</EmailVerified>
      <Remark>adfasdf</Remark>
    </User>
  </CreateUserResult>
  <ResponseMetadata>
    <RequestId>7a62c49f-347e-4fc4-9331-6e8eEXAMPLE</RequestId>
  </ResponseMetadata>
</CreateUserResponse>
```

- JSON示例

```
{
  "RequestId": "7a62c49f-347e-4fc4-9331-6e8eEXAMPLE",
  "User": {
    "UserId": "待确定类型和位数",
    "Path": "/division_abc/subdivision_xyz/",
    "UserName": "Bob",
    "Krn": "krn:ksc:iam::12345678901234567890:user/division_abc/subdivision_xyz/Bob",
    "RealName": "金山云",
    "CreateDate": "2015-11-01T12:00:00Z",
    "Phone": "13300001111",
    "Email": "asdfsaf@sd.com",
    "PhoneVerified": "Yes",
    "EmailVerified": "No",
    "Remark": "adfasdf"
  }
}
```

删除子用户

限制说明

直接调用删除用户的openAPI操作时, 只有用户无任何关联策略、无任何AccessKey、未设置登录密码才能被删除; 如果在控制台操作删除用户, 则控制台负责调用相关openAPI解除上述关联或者删除上述信息后再删除用户。

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是	确定值	[a-zA-Z]+	操作接口名, 系统规定参数, 取值: DeleteUser
UserName	String	是	最短1, 最长64	[\w+,.@-]+	待删除的IAM用户的用户名

错误码（调用不成功）

错误代码 (Code)	错误消息 (Message)	HTTP 状态码	中文描述（语义）
InvalidParameterValue	An value must be supplied for the input parameter %s.	400	输入参数%s的值不能为空，具体参数名在%s说明
InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter %s.	400	输入参数 %s的值无效、不合法或者超出范围，具体参数名在%s说明
DeleteConflict	The request was rejected because it attempted to delete a resource that has attached subordinate entities	409	无法删除拥有附属实体的资源
UserNoSuchEntity	The request was rejected because it referenced 'User' that does not exist.	404	请求被拒绝，因其引用的“用户”不存在

示例

请求示例

https://iam.api.ksyun.com/?Action=DeleteUser&UserName=Bob&公共请求参数

返回示例

- XML格式

```
<DeleteUserResponse>
  <ResponseMetadata>
    <RequestId>bc1a2796-c37a-4b03-8398-ba2281e48a3e</RequestId>
  </ResponseMetadata>
</DeleteUserResponse>
```

- JSON示例

```
{
  "ResponseMetadata": {
    "RequestId": "9b89c62d-64fa-470c-8a41-69326c0e95a7"
  }
}
```

查询子用户基本信息

查询子用户基本信息（GetUser）

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是	确定值	[a-zA-Z]+	操作接口名，系统规定参数，取值：GetUser
UserName	String	否	最短1, 最长64	[\w+=,.\@-]+	待查询的IAM用户名，未指定调用时会依据AKId推测用户名

返回元素（调用成功）

名称	类型	描述
User	UserType	待查询用户的基本信息

错误码（调用不成功）

错误代码 (Code)	错误消息 (Message)	HTTP 状态码	中文描述（语义）
UserNoSuchEntity	The request was rejected because it referenced 'User' that does not exist.	404	请求被拒绝，因其引用的“用户”不存在

示例

请求示例

https://iam.api.ksyun.com/?Action=GetUser&UserName=Bob&公共请求参数

返回示例

- XML格式

```
<GetUserResponse>
  <GetUserResult>
    <User>
      <Path>/</Path>
      <UserName>Bob</UserName>
      <RealName/>
      <UserId>IPwYlmqdSUWDwEGxDgM09Q</UserId>
      <Krn>krn:ksc:iam::73407420:user/Bob</Krn>
      <CreateDate>2016-05-16T08:46:01.000Z</CreateDate>
      <PasswordLastUsed/>
    </User>
  </GetUserResult>
  <ResponseMetadata>
    <RequestId>069a742f-5846-4e26-8ec5-2ea1cdad60ea</RequestId>
  </ResponseMetadata>
</GetUserResponse>
```

- JSON示例

```
{
  "GetUserResult": {
    "User": {
      "Path": "/",
      "UserName": "Bob",
      "UserId": "IPwYlmqdSUWDwEGxDgM09Q",
      "Krn": "krn:ksc:iam::73407420:user/Bob",
      "CreateDate": "2016-05-16T08:46:01.000Z"
    }
  },
  "ResponseMetadata": {
    "RequestId": "98a7e264-4a8f-4426-b4db-19972e863bed"
  }
}
```

更新子用户基本信息

更新子用户的基本信息

注：更新用户的基本信息可能会导致IAM的访问控制策略失效。

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是	确定值	[a-zA-Z]+	操作接口名，系统规定参数，取值：UpdateUser
NewPath	String	否	最短1，最长512	/	可选参数，用户名的路径，当前只能传入/
NewUserName	String	否	最短1，最长64	[\w+,.@-]+	可选参数，新的IAM用户名，如果未传入则说明不改变
NewRealName	String	否	最短2，最长128	[\u4e00-\u9fff]+	可选参数，新的IAM用户的真实姓名，如果未传入则说明不改变
NewEmail	String	是	最短2，最长128		可选参数，待更新的IAM用户Email
NewPhone	String	是	11个字符	1[34578]\d{9}	可选参数，待更新的IAM用户手机

返回元素（调用成功）

名称	类型	描述
User	UserType	更新后用户的基本信息

错误码（调用不成功）

错误代码(Code)	错误消息(Message)	HTTP 状态码	中文描述（语义）
InvalidParameterValue	An value must be supplied for the input parameter %s.	400	输入参数%s的值不能为空，具体参数名在%s说明

InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter %s.	400	输入参数 %s 的值无效、不合法或者超出范围，具体参数名在 %s 说明
UserNotFoundException	The request was rejected because it referenced a 'User' that does not exist.	404	请求被拒绝，因其引用的“用户”不存在

示例（调用成功）

请求示例

http://iam.api.ksyun.com?Action=UpdateUser&UserName=Bob&NewUserName=test2&&公共请求参数

返回示例

- XML格式

```
<UpdateUserResponse>
  <UpdateUserResult>
    <User>
      <UserId>x-g</UserId>
      <UserName>test2</UserName>
      <RealName></RealName>
      <CreateDate>2017-06-20T02:22:43Z</CreateDate>
      <Phone>1234567890</Phone>
      <Email>test2@test.com</Email>
      <PhoneVerified>No</PhoneVerified>
      <EmailVerified>No</EmailVerified>
      <Krn>krn:ksc:iam::xxxx:user/test2</Krn>
      <PasswordResetRequired></PasswordResetRequired>
    </User>
  </UpdateUserResult>
  <ResponseMetadata>
    <RequestId>483b7316-61c5-452c-9a25-56af8d293e4d</RequestId>
  </ResponseMetadata>
</UpdateUserResponse>
```

- JSON示例

```
{
  "UpdateUserResult": {
    "User": {
      "UserId": "x-g",
      "UserName": "test2",
      "RealName": "",
      "CreateDate": "2017-06-20T02:22:43Z",
      "Phone": "1234567890",
      "Email": "test2@test.com",
      "PhoneVerified": "No",
      "EmailVerified": "No",
      "Krn": "krn:ksc:iam::xxxx:user/test2",
      "PasswordResetRequired": false
    }
  },
  "ResponseMetadata": {
    "RequestId": "ead3df48-5681-45fb-89eb-a78f0ca85df8"
  }
}
```

查询子用户列表

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是	确定值	[a-zA-Z]+	操作接口名，系统规定参数，取值：ListUsers
PathPrefix	String	否	最短1，最长512	/	可选参数，用户名的路径，当前只能传入/
Marker	String	否	最短1，最长320	[:graph:]+	当返回结果被截断时，使用本次返回的Marker值用于标示下次调用查询用户操作的起始点
MaxItems	Integer	否	最小值1，最大值1000，默认100		用于限制本次查询结果返回的用户数量，如果仍有额外用户未显示，则返回元素中的IsTruncated的值会被设置为true，同时返回下次查询的起始点Marker值

返回元素（调用成功）

名称	类型	描述
IsTruncated	Boolean	截断标志位，用于说明是否仍有更多的用户未显示全
Marker	String	如果截断标志位是true，Marker值被设定，用于标识下次查询的起始点
Users	UserType List	用户基本信息列表

错误码（调用不成功）

错误代码 (Code)	错误消息 (Message)	HTTP 状态码	中文描述（语义）
InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter %s.	400	输入参数 %s 的值无效、不合法或者超出范围，具体参数名在 %s 说明

示例（调用成功）

请求示例

https://iam.api.ksyun.com/?Action=ListUsers&公共请求参数

返回示例

• XML格式

```
<ListUsersResponse>
  <ListUserResult>
    <Users>
      <member>
        <Id>1234</Id>
        <Path>/</Path>
        <UserName>Bob</UserName>
        <RealName>测试</RealName>
        <UserId>1X9TJiSfTtq-rAJ3SVtcow</UserId>
        <Krn>krn:ksc:iam::73407420:user/Bob</Krn>
        <CreateDate>2016-05-16T14:44:20.000Z</CreateDate>
        <PasswordLastUsed/>
      </member>
    </Users>
    <Marker>MTE0Y2Y1MWEtNmQxZC00YjU3LTk5NzgtZTkzMjA2NDRkNjE0XzE=</Marker>
    <IsTruncated>true</IsTruncated>
  </ListUserResult>
  <ResponseMetadata>
    <RequestId>114cf51a-6d1d-4b57-9978-e9320644d614</RequestId>
  </ResponseMetadata>
</ListUsersResponse>
```

• JSON示例

```
{
  "ListUserResult": {
    "Users": {
      "member": [
        {
          "Id": "1234",
          "Path": "/",
          "UserName": "Bob",
          "RealName": "测试",
          "UserId": "1X9TJiSfTtq-rAJ3SVtcow",
          "Krn": "krn:ksc:iam::73407420:user/Bob",
          "CreateDate": "2016-05-16T14:44:20.000Z"
        }
      ]
    },
    "Marker": "YT11ZDk10GUtYTVkNS00NjU4LWE1NDItMjc4MwY2NTVhMG13XzE=",
    "IsTruncated": true
  },
  "ResponseMetadata": {
    "RequestId": "a9ed958e-a5d5-4658-a542-2781f655d0b7"
  }
}
```

新建子用户登录配置

接口描述

新建子用户登录配置，登录配置包括登录密码和是否需要下次登录设置新密码，只有拥有登录配置的用户才能登录控制台，成功新建用户登录配置，会返回用户名、密码创建时间和是否需要下次登录设置密码，如果无法新建则返回相关报错信息。

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是	确定值	[a-zA-Z]+	操作接口名，系统规定参数，取值：CreateLoginProfile
Password	String	是	最短1，最长128	[:graph:]+	用户登录密码
UserName	String	是	最短1，最长64	[\w+,.@-]+	待创建登录配置的IAM用户名
PasswordResetRequired	Boolean	否		无	可选参数，标识用户下次成功登录后是否需要设置新密码，带参数调用即为true，否则不带参数为false

返回元素（调用成功）

名称	类型	描述
LoginProfile	LoginProfileType	用户登录配置信息

错误码（调用不成功）

错误代码(Code)	错误消息(Message)	HTTP 状态码	中文描述（语义）
InvalidParameterValue	An value must be supplied for the input parameter %s.	400	输入参数%s的值不能为空，具体参数名在%s说明
InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter %s.	400	输入参数 %s的值无效、不合法或者超出范围，具体参数名在%s说明
UserNoSuchEntity	The request was rejected because it referenced 'User' that does not exist.	404	请求被拒绝，因其引用的“用户”不存在
EntityAlreadyExists	The request was rejected because it attempted to create a resource that already exists	409	请求被拒绝，因待创建资源已经存在
PasswordPolicyViolation	The request was rejected because the provided password did not meet the requirements imposed by the account password policy	400	登录密码不符合用户账户的密码策略

示例（调用成功）

请求示例

http://iam.api.ksyun.com/?Action=CreateLoginProfile&Password=123456&UserName=Bob&公共请求参数

返回示例

- XML格式

```
<CreateLoginProfileResponse>
  <CreateLoginProfileResult>
    <LoginProfile>
      <PasswordResetRequired>>false</PasswordResetRequired>
      <UserName>Bob</UserName>
      <CreateDate>2016-05-16T15:28:23.379Z</CreateDate>
    </LoginProfile>
  </CreateLoginProfileResult>
  <ResponseMetadata>
    <RequestId>219f48ee-080c-4981-8a11-f6f904dad3b0</RequestId>
  </ResponseMetadata>
</CreateLoginProfileResponse>
```

- JSON示例

```
{
  "CreateLoginProfileResult": {
    "LoginProfile": {
```

```

    "PasswordResetRequired": "false",
    "UserName": "Bob",
    "CreateDate": "2016-05-16T15:29:49.136Z"
  },
  "ResponseMetadata": {
    "RequestId": "2cc173e6-5e99-46a1-9ebc-c13149222640"
  }
}

```

删除子用户登录配置

删除子用户登录配置并不能阻止用户通过openAPI访问，要彻底阻止用户访问，需要禁用/删除用户的AccessKey。

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是	确定值	[a-zA-Z]+	操作接口名，系统规定参数，取值：DeleteLoginProfile
UserName	String	是	最短1, 最长64	[\w+=,.\@-]+	待删除登录配置的IAM用户名

返回元素（调用成功）

无

错误码（调用不成功）

错误代码(Code)	错误消息(Message)	HTTP 状态码	中文描述（语义）
InvalidParameterValue	An value must be supplied for the input parameter %s.	400	输入参数%s的值不能为空，具体参数名在%s说明
InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter %s.	400	输入参数 %s的值无效、不合法或者超出范围，具体参数名在%s说明
UserNoSuchEntity	The request was rejected because it referenced 'User' that does not exist.	404	请求被拒绝，因其引用的“用户”不存在

示例（调用成功）

请求示例

https://iam.api.ksyun.com/?Action=DeleteLoginProfile&UserName=Bob&公共请求参数

返回示例

- XML格式

```

<DeleteLoginProfileResponse>
  <ResponseMetadata>
    <RequestId>502304be-a748-4ebf-b5af-7406d1e031c0</RequestId>
  </ResponseMetadata>
</DeleteLoginProfileResponse>

```

- JSON示例

```

{
  "ResponseMetadata": {
    "RequestId": "e2b2ae00-ed4c-4471-b8c9-c8b74eb76895"
  }
}

```

查询子用户登录配置

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是	确定值	[a-zA-Z]+	操作接口名，系统规定参数，取值：GetLoginProfile
UserName	String	是	最短1, 最长64	[\w+=,.\@-]+	待查询登录配置的IAM用户名

返回元素（调用成功）

名称	类型	描述
LoginProfile	LoginProfileType	用户登录配置相关信息

错误码（调用不成功）

错误代码 (Code)	错误消息 (Message)	HTTP 状态码	中文描述（语义）
InvalidParameterValue	An value must be supplied for the input parameter %s.	400	输入参数%s的值不能为空，具体参数名在%s说明
InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter %s.	400	输入参数 %s的值无效、不合法或者超出范围，具体参数名在%s说明
UserNoSuchEntity	The request was rejected because it referenced 'User' that does not exist.	404	请求被拒绝，因其引用的“用户”不存在
LoginProfileNoSuchEntity	The request was rejected because it referenced 'LoginProfile' that does not exist.	404	请求被拒绝，因其引用的“登录配置文件”不存在

示例（调用成功）

请求示例

https://iam.api.ksyun.com/?Action=GetLoginProfile&UserName=Bob&公共请求参数

返回示例

- XML格式

```
<GetLoginProfileResponse>
  <GetLoginProfileResult>
    <LoginProfile>
      <PasswordResetRequired>false</PasswordResetRequired>
      <UserName>Bob</UserName>
      <CreateDate>2016-05-16T15:27:10.000Z</CreateDate>
    </LoginProfile>
  </GetLoginProfileResult>
  <ResponseMetadata>
    <RequestId>e5c74f82-0d97-4245-8fba-03a40d92bf7e</RequestId>
  </ResponseMetadata>
</GetLoginProfileResponse>
```

- JSON示例

```
{
  "CreateLoginProfileResult": {
    "LoginProfile": {
      "PasswordResetRequired": "false",
      "UserName": "ziyao",
      "CreateDate": "2016-05-16T15:27:10.000Z"
    }
  },
  "ResponseMetadata": {
    "RequestId": "a4f12801-f61d-4142-b45f-240455291c00"
  }
}
```

更新子用户登录配置

注：如果是单纯修改子用户自己的登录密码，调用[ChangePassword](#)接口。

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是	确定值	[a-zA-Z]+	操作接口名，系统规定参数，取值：UpdateLoginProfile
Password	String	否	最短1，最长128	[:graph:]+	可选参数，用户的新登录密码

UserName	String	是	最短1, 最长64	[\w+=,.\@-]+	待更新登录配置的IAM用户名
PasswordResetRequired	Boolean	否	无		可选参数, 标识用户下次成功登录后是否需要设置新密码, 带参数调用即为true, 否则不带参数为false
OpenLoginProtection	Boolean	否	无		可选参数, 开启登录操作保护, 不带参数不处理, 带参数为true开启, false关闭
OpenSecurityProtection	Boolean	否	无		可选参数, 开启敏感操作保护, 不带参数不处理, 带参数为true开启, false关闭
ViewAllProject	Boolean	否	无		可选参数, 标识用户是否可以查看全部项目, 带参数调用即为true, 否则不带参数为false

返回元素（调用成功）

无

错误码（调用不成功）

错误代码 (Code)	错误消息 (Message)	HTTP 状态码	中文描述 (语义)
InvalidParameterValue	An value must be supplied for the input parameter %s.	400	输入参数%s的值不能为空, 具体参数名在%s说明
InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter %s.	400	输入参数 %s的值无效、不合法或者超出范围, 具体参数名在%s说明
UserNoSuchEntity	The request was rejected because it referenced 'User' that does not exist.	404	请求被拒绝, 因其引用的“用户”不存在
LoginProfileNoSuchEntity	The request was rejected because it referenced 'LoginProfile' that does not exist.	404	请求被拒绝, 因其引用的“登录配置文件”不存在
PasswordPolicyViolation	The request was rejected because the provided password did not meet the requirements imposed by the account password policy.	400	登录密码不符合用户账户的密码策略

示例（调用成功）

请求示例

https://iam.api.ksyun.com/?Action=UpdateLoginProfile&Password=123456&UserName=Bob&公共请求参数

返回示例

- JSON示例

```
{
  "CreateLoginProfileResult": {
    "LoginProfile": {
      "UserName": "user01",
      "OpenLoginProtection": true,
      "OpenSecurityProtection": false,
      "PasswordResetRequired": true,
      "UpdateDate": "2019-11-21T10:10:32Z",
      "Password": "123456789"
    }
  },
  "RequestId": "64c8078d-b818-41b5-aeel-d16af7f513f3"
}
```

修改子用户密码

注1: 这个接口是子用户修改自身密码的接口, 金山云主账户调用报错。

注2: 如果在AccountPasswordPolicy的AllowUsersToChangePassword被设置为true时, 则所有用户都可以修改自己的密码; 如果AccountPasswordPolicy的AllowUsersToChangePassword被设置为false, 则用户能否修改自己的密码需要看是否被授予相关权限 (ChangePassword); 这个业务逻辑对于IAM用户登录时需要修改密码一样适用, 即如果IAM用户登录时被要求修改密码 (LoginProfile设置), 如果用户没有被授权权限, 则无法修改密码也无法完成登录。

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是	确定值	[a-zA-Z]+	操作接口名, 系统规定参数, 取值: ChangePassword
OldPassword	String	是	最短1, 最长128	[:graph:]+	用户的旧登录密码
NewPassword	String	是	最短1, 最长128	[:graph:]+	用户的新登录密码

返回元素 (调用成功)

无

错误码 (调用不成功)

错误代码 (Code)	错误消息 (Message)	HTTP 状态码	中文描述 (语义)
InvalidParameterValue	An value must be supplied for the input parameter %s.	400	输入参数%s的值不能为空, 具体参数名在%s说明
PasswordPolicyViolation	The request was rejected because the provided password did not meet the requirements imposed by the account password policy	400	新登录密码不符合用户账户的密码策略
AccessDenied	The request was rejected because the old password was incorrect	409	请求被拒绝, 因旧密码不正确

示例 (调用成功)

请求示例

https://iam.api.ksyun.com/?Action=ChangePassword&OldPassword=123456&NewPassword=654321&公共请求参数

返回示例

- XML格式

```
<ChangePasswordResponse>
  <ResponseMetadata>
    <RequestId>502304be-a748-4ebf-b5af-7406d1e031c0</RequestId>
  </ResponseMetadata>
</ChangePasswordResponse>
```

- JSON示例

```
{
  "ResponseMetadata": {
    "RequestId": "e2b2ae00-ed4c-4471-b8c9-c8b74eb76895"
  }
}
```

获取子用户登录态

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是	确定值	[a-zA-Z]+	操作接口名, 系统规定参数, 取值: GetUserSession
UserName	String	否	最短1, 最长64	[\w+\.,@-]+	等待获取登录态IAM用户名, 未指定调用时会依据AKId推测用户名
EmbeddedSession	String	否	最短2, 最长3	[Yes No]	返回的Session是否仅在Embedded控制台有效, 默认是No
Duration	int	否	最小60, 最大900	int	返回连接有效时间, 默认60s
CallbackType	String	否	-	[REDIRECT JSONP]	返回url的类型, 返回支持302的重定向和JSONP的访问。默认302从定向
CallbackUrl	String	否	4096	url地址	如果是REDIRECT模式, 生成callback地址

返回元素 (调用成功)

名称	类型	描述
UserSession	UserType	登录Session信息

错误码（调用不成功）

错误代码(Code)	错误消息(Message)	HTTP 状态码	中文描述（语义）
UserBadGenerationSessionFail	to generate the login session.	500	服务器生成用户session失败

示例

请求示例

https://iam.api.ksyun.com/?Action=GetUserSession&UserName=Bob&公共请求参数

返回示例

- XML格式

```
<GetUserSessionResponse>
  <GetUserSessionResult>
    <Url>http://embedded.ksyun.com/checksession?AccountId=73406424&Code=872579651</Url>
    <Expiration>2017-09-04T11:12:49Z</Expiration>
  </GetUserSessionResult>
  <ResponseMetadata>
    <RequestId>67113f08-24da-41f2-8130-8bbf1eeb4295</RequestId>
  </ResponseMetadata>
</GetUserSessionResponse>
```

- JSON示例

```
{
  "GetUserSessionResult": {
    "Url": "http://embedded.ksyun.com/checksession?AccountId=73406424&Code=2227559128",
    "Expiration": "2017-09-04T11:12:20Z"
  },
  "RequestId": "3c1221a9-debe-45ee-9d92-cc76acd04119"
}
```

创建成员账号

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是	确定值	[a-zA-Z]+	操作接口名，系统规定参数，取值：CreateAccount
AccountName	String	是	最短6，最长18	^[a-zA-Z][a-zA-Z0-9]+	欲创建的账号名
Password	String	是	最短8，最长32	string	欲创建的账号密码，无空格
Mobile	string	是	最小60，最大900	int	欲创建的账号的手机号，国内手机号码
SecureEmail	String	否	最长64	string	欲创建的账号的安全邮箱

返回元素（调用成功）

名称	类型	描述
CreateAccountResult	AccountType	账号信息

错误码（调用不成功）

错误代码(Code)	错误消息(Message)	HTTP 状态码	中文描述（语义）
AccountNameAlreadyExists	The request was rejected because it attempted to create a AccountName that already exists.	409	账号名已经存在

AccountInvalidParameter	The username field is required.	400	参数错误
AccountInternalError	Call passport service error.	400	调用Passport服务返回的其他类型错误

示例

请求示例

<https://iam.api.ksyun.com/?Action=CreateAccount&Version=2015-11-01&AccountName=test&Password=test123&Mobile=18612345678&SecureEmail=test@163.com>

返回示例

- XML格式

```
<CreateAccountResponse>
<CreateAccountResult>
  <AccountId>2579651</AccountId>
  <Mobile>185*****08</Mobile>
  <AccountName>test</AccountName>
  <SecureEmail>t*st@163.com</SecureEmail>
  <AccessKeyId>lwelsolsdkjf</AccessKeyId>
  <SecretAccessKey>sdfsdfsdfsdf</SecretAccessKey>
</CreateAccountResult>
<ResponseMetadata>
  <RequestId>67113f08-24da-41f2-8130-8bbf1eeb4295</RequestId>
</ResponseMetadata>
</GetUserSessionResponse>
```

- JSON示例

```
{
  "CreateAccountResult": {
    "AccountId": "2579651",
    "Mobile": "185*****08",
    "AccountName": "test",
    "SecureEmail": "t*st@163.com",
    "AccessKeyId": "lwelsolsdkjf",
    "SecretAccessKey": "sdfsdfsdfsdf"
  },
  "RequestId": "3c1221a9-debe-45ee-9d92-cc76acd04119"
}
```

获取子用户AK最后使用时间

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是	确定值	[a-zA-Z]+	操作接口名，系统规定参数，取值：ListAllUserAccessKeys

返回元素（调用成功）

名称	类型	描述
AccessKeyList	AccessKey	List 子用户AccessKey最后使用时间列表

示例（调用成功）

请求示例

<https://iam.api.ksyun.com/?Action=ListAllUserAccessKeys&公共请求参数>

返回示例

- JSON示例

```
{
```

```

"AccessKeyList": [
  {
    "UserName": "test_ak",
    "AccessKey": "AKLT0dpEERKsSG6mIw",
    "AkLastUsedTime": "2022-07-14 16:06:22"
  },
  {
    "UserName": "testmfa222",
    "AccessKey": "AKLTCjVDJUv6RfqrQ",
    "AkLastUsedTime": "-"
  },
  {
    "UserName": "tess",
    "AccessKey": "AKLTlw-ehPucSN",
    "AkLastUsedTime": "2022-07-14 16:00:22"
  }
],
"RequestId": "4a1c650-ef28-4c4f-be15-9ec425be60fb"
}

```

生成访问密钥

注1: 金山云账户（即主账户）可以为自己或者任意子用户生成访问密钥，授权的子用户可为任意子用户（非主账户）生成访问密钥。

注2: 为保证账户安全，私有访问密钥只有在创建时返回/显示一次，请用户自己妥善保管私有访问密钥，如果丢失则无法恢复，只能删除并重新生成访问密钥。

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是	确定值	[a-zA-Z]+	操作接口名，系统规定参数，取值：CreateAccessKey
UserName	String	否	最短1, 最长64	[\w+=,.\@-]+	待生成访问密钥的IAM用户名

返回元素（调用成功）

名称	类型	描述
AccessKey	AccessKeyType	新创建的访问密钥信息

错误码（调用不成功）

错误代码(Code)	错误消息(Message)	HTTP 状态码	中文描述（语义）
InvalidParameterValue	An value must be supplied for the input parameter %s.	400	输入参数%s的值不能为空，具体参数名在%s说明
InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter %s.	400	输入参数 %s的值无效、不合法或者超出范围，具体参数名在%s说明
UserAccessKeyIdExceedsLimit	The request was rejected because it attempted to create access key to the user that beyond the current IAM User Access Key Count limits.	409	请求被拒绝，因其欲为用户创建的访问密钥数量超过当前账户限制(AccessKey限制为2)

示例（调用成功）

请求示例

https://iam.api.ksyun.com/?Action=CreateAccessKey&UserName=Bob&公共请求参数

返回示例

- XML格式

```

<CreateAccessKeyResponse>
  <CreateAccessKeyResult>
    <AccessKey>
      <UserName>Bob</UserName>
      <AccessKeyId>AKLTrqC01brvRhCfETDCFx1LXA</AccessKeyId>
      <SecretAccessKey>ABCDE12345HyHr0BEB5PvMg3174aL0iFhQdVzykDmqUS/025XjSxCoNDUcguG7ms3g==</SecretAccessKey>
    </AccessKey>
  </CreateAccessKeyResult>
</CreateAccessKeyResponse>

```

```

    <Status>Active</Status>
    <CreateDate>2016-05-16T16:33:44.363Z</CreateDate>
  </AccessKey>
</CreateAccessKeyResult>
<ResponseMetadata>
  <RequestId>f63e2a04-8f10-4509-b04f-d3c72eb1d402</RequestId>
</ResponseMetadata>
</CreateAccessKeyResponse>

```

- JSON示例

```

{
  "CreateAccessKeyResult": {
    "AccessKey": {
      "UserName": "Bob",
      "AccessKeyId": "AKLTqC01brvRhCFETDCFX1LXA",
      "SecretAccessKey": "ABCDE12345HyHr0BEB5PvmG3I74aL0iFhQdVzykDmqUS/025XjSxCoNDUcgU7ms3g==",
      "Status": "Active",
      "CreateDate": "2016-05-16T16:36:17.397Z"
    }
  },
  "ResponseMetadata": {
    "RequestId": "78799d1b-9a17-4ae3-93f8-c80f283554e8"
  }
}

```

删除访问密钥

注：直接调用接口删除，状态为启用的AK也能删除，建议请先禁用AK，确保业务线不受影响再调用接口操作删除

请求参数

名称	类型	必填	描述
Action	String	是	公共参数，本接口取值：DeleteAccessKey
UserName	String	是	IAM子用户的用户名
AccessKeyId	String	是	访问密钥ID

返回元素（调用成功）

名称	类型	描述
result	Boolean	示例值：true
RequestId	String	示例值：a0ddd4d1-a30b-487a-9177-ef851ccf685f

错误码（调用不成功）

错误代码(Cod e)	错误消息(Message)	HTTP 状态码	中文描述（语义）
InvalidParameterValue	An value must be supplied for the input parameter %s.	400	输入参数%s的值不能为空，具体参数名在%s说明
InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter %s.	400	输入参数 %s的值无效、不合法或者超出范围，具体参数名在%s说明

示例（调用成功）

请求示例

```

curl -X GET|POST
' http://iam.api.ksyun.com/?Action=DeleteAccessKey&Version=2015-11-01'
-H 'authorization: xxx, SignedHeaders=content-type;host;x-amz-date, Signature=xxx'
-H 'cache-control: no-cache'
-H 'content-type: form/x-www-form-urlencoded'
-H 'host: iam.ksyun.com.com'
-H 'x-amz-date: 20170720T022802Z'
-d '{"UserName": "xxx", "AccessKeyId": "xxx"}'

```

返回示例

- JSON示例

```
{
  "result":
  true,
  "RequestId":
  "a0ddd4d1-a30b-487a-9177-ef851ccf685f"
}
```

查询访问密钥列表

注1: 金山云账户（即主账户）可以查询自己或任意IAM用户的访问密钥列表，授权的子用户可查询任意IAM用户（非主账户）访问密钥列表。

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是	确定值	[a-zA-Z]+	操作接口名，系统规定参数，取值：ListAccessKeys
UserName	String	否	最短1，最长64	[\w+=,.@-]+	待查询访问密钥列表的子用户名

返回元素（调用成功）

名称	类型	描述
AccessKeyMetadata	AccessKeyMetadata_List_Type	访问密钥元数据列表

错误码（调用不成功）

错误代码 (Code)	错误消息 (Message)	HTTP 状态码	中文描述（语义）
InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter %s.	400	输入参数 %s 的值无效、不合法或者超出范围，具体参数名在 %s 说明
UserAlreadyExists	User with name %s already exists.	409	用户名 %s 的用户已经存在，具体用户名在 %s 说明

示例（调用成功）

请求示例

https://iam.api.ksyun.com/?Action=ListAccessKeys&UserName=Bob&公共请求参数

返回示例

- XML格式

```
<ListAccessKeysResponse>
  <ListAccessKeyResult>
    <AccessKeyMetadata>
      <member>
        <UserName>Bob</UserName>
        <AccessKeyId>AKLT0Psa0tFgS1KmF7zM3KqAmA</AccessKeyId>
        <Status>Active</Status>
        <CreateDate>2016-05-16T16:36:17.000Z</CreateDate>
      </member>
      <member>
        <UserName>Bob</UserName>
        <AccessKeyId>AKLTa0pqvEVBQfuqjz-oxUgooA</AccessKeyId>
        <Status>Active</Status>
        <CreateDate>2016-05-16T16:47:49.000Z</CreateDate>
      </member>
    </AccessKeyMetadata>
  </ListAccessKeyResult>
  <ResponseMetadata>
    <RequestId>64f5bac9-49cc-4035-aa9b-558dd2e64cbe</RequestId>
  </ResponseMetadata>
</ListAccessKeysResponse>
```

- JSON示例

```
{
  "ListAccessKeyResult": {
    "AccessKeyMetadata": {
```



```

    "member": [
      {
        "UserName": "Bob",
        "AccessKeyId": "AKLT0Psa0tFgS1KmF7zM3KqAmA",
        "Status": "Inactive",
        "CreateDate": "2016-05-16T16:36:17.000Z"
      },
      {
        "UserName": "Bob",
        "AccessKeyId": "AKLTa0ppqVEVBQfuqjz-oxUgooA",
        "Status": "Active",
        "CreateDate": "2016-05-16T16:47:49.000Z"
      }
    ]
  },
  "ResponseMetadata": {
    "RequestId": "b5ec6ba4-5f1d-489f-83bc-7021be90ffb6"
  }
}

```

更新访问密钥

为主账户和子用户修改指定访问密钥状态

注1: 金山云账户（即主账户）可以修改自己或任意子用户的访问密钥状态，授权的子用户可为任意子用户（非主账户）修改访问密钥状态。

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是	确定值	[a-zA-Z]+	操作接口名，系统规定参数，取值：DeleteAccessKey
AccessKeyId	String	是	最短20, 最长32	AKLT[a-zA-Z0-9-]+	待删除的访问密钥ID
Status	String	是	6或8	Active/Inactive	修改后的访问密钥的状态，Active为可用，Inactive为不可用
UserName	String	否	最短1, 最长64	[\w+=, .@-]+	待删除访问密钥的子用户名

返回元素（调用成功）

无

错误码（调用不成功）

错误代码 (Code)	错误消息 (Message)	HTTP 状态码	中文描述（语义）
InvalidParameterValue	An value must be supplied for the input parameter %s.	400	输入参数%s的值不能为空，具体参数名在%s说明
InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter %s.	400	输入参数 %s的值无效、不合法或者超出范围，具体参数名在%s说明
UserNoSuchEntity	The request was rejected because it referenced 'User' that does not exist.	404	请求被拒绝，因其引用的“用户”不存在
AccessKeyNoSuchEntity	The request was rejected because it referenced 'AccessKey' that does not exist.	404	请求被拒绝，因其引用的“访问密钥”不存在

示例（调用成功）

请求示例

https://iam.api.ksyun.com/?Action=UpdateAccessKey&UserName=Bob&AccessKeyId=AKLT1234567890ABCDEFGHIJ&Status=Inactive&公共请求参数

返回示例

- XML格式

```

<UpdateAccessKeyResponse>
  <ResponseMetadata>

```

```
<RequestId>a8dcc690-7c86-470f-8959-87c03f7d8cd8</RequestId>
</ResponseMetadata>
</UpdateAccessKeyResponse>
```

- JSON示例

```
{
  "ResponseMetadata": {
    "RequestId": "502304be-a748-4ebf-b5af-7406d1e031c0"
  }
}
```

新建策略

注1: 策略是授权文档，其属主是金山云账户（即主账户），通过[AttachUserPolicy](#)操作将策略中描述的权限授予相应的子用户，被授权的用户在访问openAPI或者控制台时会受到策略中所描述的权限的制约。

注2: 除了账户自己创建的自定义策略外，IAM系统还为每个主账户提供内置的全局系统策略，全局系统策略不能被账户更改，但可以被所有账户使用，自定义策略可以与系统策略同名（因为PolicyKrn是不同的）。

注3: PolicyId是IAM系统内部唯一标识一个策略的Id，22字符。

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是	确定值	[a-zA-Z]+	操作接口名，系统规定参数，取值：CreatePolicy
Description	String	否	最短0, 最长1000	中英文可见字符	可选参数，策略描述-用自然语言说明定义在策略中的权限，策略描述设定后不可改变, 支持中文字符
Path	String	否	最短1, 最长512	/	可选参数，策略的路径，如果未传入则默认值是/
PolicyName	String	是	最短1, 最长128	[\w+.,@-]+	待创建策略的策略名称
PolicyDocument	String	是	最短1, 最长5K	[\w\s:~*{}[\]/\$-]+	待创建的自定义策略的策略文档内容

返回元素（调用成功）

名称	类型	描述
Policy	PolicyType	新创建策略的元数据信息

错误码（调用不成功）

错误代码(Code)	错误消息(Message)	HTTP 状态码	中文描述（语义）
InvalidParameterValue	An value must be supplied for the input parameter %s.	400	输入参数%s的值不能为空，具体参数名在%s说明
InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter %s.	400	输入参数 %s的值无效、不合法或者超出范围，具体参数名在%s说明
PolicyAlreadyExists	The request was rejected because it attempted to create a policy that already exists	409	请求被拒绝，因待创建策略已经存在
PolicyLimitExceeded	The request was rejected because it attempted to create policy beyond the current KSC account limits.	409	请求被拒绝，因其创建的策略数量超过当前账户限制(Policy数量限制50)
PolicyDocumentNotJsonFormat	The 'PolicyDocument' is not json format.	400	策略文档不是Json格式
PolicyDocumentInvalidVersion	The 'Version' in 'PolicyDocument' is invalid.	400	策略文档中的Version元素无效
PolicyDocumentInvalidStatement	The 'Statement' in 'PolicyDocument' is invalid.	400	策略文档中的Statement元素无效
PolicyDocumentInvalidEffect	The 'Effect' in 'PolicyDocument' is invalid.	400	策略文档中的Effect元素无效
PolicyDocumentInvalidResource	The 'Resource' in 'PolicyDocument' is invalid.	400	策略文档中的Resource元素无效

PolicyDocumentInvalidAction The 'Action' in 'PolicyDocument' is invalid. 400 策略文档中的Action元素无效

示例（调用成功）

请求示例

```
https://iam.api.ksyun.com/?Action=CreatePolicy
&PolicyDocument={"Version":"2015-11-01","Statement":[{"Effect":"Allow","Action":"iam:*",
"Resource":"*"}]}
&PolicyName=IAMAdmin
&公共请求参数
```

返回示例

- XML格式

```
<CreatePolicyResponse>
  <CreatePolicyResult>
    <Policy>
      <PolicyId>ZRE0yXBkTx2juW1JT-htWg</PolicyId>
      <DefaultVersionId>v1</DefaultVersionId>
      <UpdateDate>2016-05-16T17:28:54.017Z</UpdateDate>
      <AttachmentCount>0</AttachmentCount>
      <Description/>
      <Path/></Path>
      <CreateDate>2016-05-16T17:28:54.017Z</CreateDate>
      <Krn>krn:ksc:iam::73407420:policy/IAMAdmin</Krn>
      <PolicyName>IAMAdmin</PolicyName>
    </Policy>
  </CreatePolicyResult>
  <ResponseMetadata>
    <RequestId>ff3ddb15-8e5b-4c23-be4b-391a997d1bad</RequestId>
  </ResponseMetadata>
</CreatePolicyResponse>
```

- JSON示例

```
{
  "CreatePolicyResult": {
    "Policy": {
      "PolicyId": "Mh20G6bfSvC07kBXf_gnXg",
      "DefaultVersionId": "v1",
      "UpdateDate": "2016-05-16T17:31:00.765Z",
      "AttachmentCount": "0",
      "Path": "/",
      "CreateDate": "2016-05-16T17:31:00.765Z",
      "Krn": "krn:ksc:iam::73407420:policy/IAMAdmin",
      "PolicyName": "IAMAdmin"
    }
  },
  "ResponseMetadata": {
    "RequestId": "85ee9446-32ee-4f9e-a3c2-b8f74f3f9686"
  }
}
```

删除策略

为主账户删除自定义策略。

注1：删除策略时需要保证策略当前不被附加到任何实体上（IAM本期只有“用户”一种实体），且除默认策略版本外，其他策略版本均已被删除，即成功删除策略流程如下：

1. 将欲删除的策略从用户上分离，使用[DetachUserPolicy](#)接口
2. 将欲删除的策略除默认版本外的其他策略版本删除，使用[DeletePolicyVersion](#)接口
3. 使用DeletePolicy接口完成策略删除动作（默认策略一并被删除）

注2：IAM系统内置的全局系统策略主账户不能删除

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
----	----	----	----------	------	----

Action	String	是	确定值	[a-zA-Z]+	操作接口名，系统规定参数，取值：DeletePolicy
PolicyKrn	String	是	最短是25，最长2048	格式：krn:ksc:iam:: <i>account-id</i> :policy/ <i>policy-name</i>	待删除策略的唯一标识

返回元素（调用成功）

无

错误码（调用不成功）

错误代码(Code)	错误消息(Message)	HTTP 状态码	中文描述（语义）
InvalidParameterValue	An value must be supplied for the input parameter %s.	400	输入参数%s的值不能为空，具体参数名在%s说明
InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter %s.	400	输入参数 %s的值无效、不合法或者超出范围，具体参数名在%s说明
PolicyDeleteConflict	The request was rejected because it attempted to delete a 'Policy' that has attached subordinate entities.	409	请求被拒绝，因其欲删除具有附加实体的策略
PolicyNoSuchEntity	The request was rejected because it referenced 'Policy' that does not exist.	404	请求被拒绝，因其引用的“策略”不存在

示例（调用成功）

请求示例

```
https://iam.api.ksyun.com?Action=DeletePolicy
&PolicyKrn=krn:ksc:iam::12345678901234567890:policy/IAMAdmin
&公共请求参数
```

返回示例

- XML格式

```
<DeletePolicyResponse>
  <ResponseMetadata>
    <RequestId>941610f5-5966-475b-a0f5-4b3102aa2619</RequestId>
  </ResponseMetadata>
</DeletePolicyResponse>
```

- JSON示例

```
{
  "ResponseMetadata": {
    "RequestId": "184816ad-ba6a-44d5-abf6-9215971778bc"
  }
}
```

查询策略元数据信息

查询主账号下指定策略的元数据信息。

注1：策略的元数据信息不包括策略授权文档的内容，如果需要查看策略授权文档内容，需要调用[GetPolicyVersion](#)接口；IAM系统内置的全局系统策略的元数据信息也可以被查询。

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是	确定值	[a-zA-Z]+	操作接口名，系统规定参数，取值：GetPolicy
PolicyKrn	String	是	最短是25，最长2048	格式：krn:ksc:iam:: <i>account-id</i> :policy/ <i>policy-name</i>	待查询策略的唯一标识

返回元素（调用成功）

名称	类型	描述
Policy	PolicyType	被查询策略的元数据信息

错误码（调用不成功）

错误代码 (Code)	错误消息 (Message)	HTTP 状态码	中文描述（语义）
InvalidParameterValue	An value must be supplied for the input parameter %s.	400	输入参数%s的值不能为空，具体参数名在%s说明
InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter %s.	400	输入参数 %s的值无效、不合法或者超出范围，具体参数名在%s说明
PolicyNoSuchEntity	The request was rejected because it referenced 'Policy' that does not exist.	404	请求被拒绝，因其引用的“策略”不存在

示例（调用成功）

请求示例

```
https://iam.api.ksyun.com/?Action=GetPolicy
&PolicyKrn=krn:ksc:iam::12345678901234567890:policy/IAMAdmin
&公共请求参数
```

返回示例

- XML格式

```
<GetPolicyResponse>
  <ResponseMetadata>
    <RequestId>c8b0eafa-80e5-4eb3-b1b8-00c5265cbce4</RequestId>
  </ResponseMetadata>
  <GetPolicyResult>
    <Policy>
      <PolicyId>eCuWLaNkTp6Pjv3j348tNw</PolicyId>
      <DefaultVersionId>v1</DefaultVersionId>
      <UpdateDate>2016-05-16T17:43:32.000Z</UpdateDate>
      <AttachmentCount>0</AttachmentCount>
      <Description/>
      <Path/></Path>
      <CreateDate>2016-05-16T17:43:32.000Z</CreateDate>
      <Krn>krn:ksc:iam::73407420:policy/IAMAdmin</Krn>
      <PolicyName>IAMAdmin</PolicyName>
    </Policy>
  </GetPolicyResult>
</GetPolicyResponse>
```

- JSON示例

```
{
  "ResponseMetadata": {
    "RequestId": "f07f9839-e89d-45f4-bc6c-de85a748aaab"
  },
  "GetPolicyResult": {
    "Policy": {
      "PolicyId": "eCuWLaNkTp6Pjv3j348tNw",
      "DefaultVersionId": "v1",
      "UpdateDate": "2016-05-16T17:43:32.000Z",
      "AttachmentCount": "0",
      "Path": "/",
      "CreateDate": "2016-05-16T17:43:32.000Z",
      "Krn": "krn:ksc:iam::73407420:policy/IAMAdmin",
      "PolicyName": "IAMAdmin"
    }
  }
}
```

查询策略信息列表

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
----	----	----	----------	------	----

Action	String	是	确定值	[a-zA-Z]+	操作接口名，系统规定参数，取值：ListPolicies
Marker	String	否	最短1，最长320	[:graph:]+	当返回结果被截断时，使用本次返回的Marker值用于标示下次调用查询策略信息列表操作的起始点
MaxItems	Integer	否	最小值1，最大值1000，默认100		用于限制本次查询结果返回的策略数量，如果仍有额外策略未显示，则返回元素中的IsTruncated的值会被设置为true，同时返回下次查询的起始点Marker值
OnlyAttached	Boolean	否			可选参数，用于过滤策略列表，如果设置则只返回已经附加到实体上策略
PathPrefix	String	否	最短1，最长512	/	可选参数，用于过滤策略列表，如果未传入则默认值是/
Scope	String	否	3或5	ALL KSC Local	用于过滤显示策略范围，ALL为全部，KSC是系统策略、Local是自定义策略

返回元素（调用成功）

名称	类型	描述
Policies	PolicyType List	被查询策略的元数据信息列表
IsTruncated	Boolean	截断标志位，用于说明是否仍有更多的策略信息未显示全
Marker	String	如果截断标志位是true，Marker值被设定，用于标识下次查询的起始点

错误码（调用不成功）

错误代码 (Code)	错误消息 (Message)	HTTP 状态码	中文描述（语义）
InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter %s.	400	输入参数 %s 的值无效、不合法或者超出范围，具体参数名在 %s 说明

示例（调用成功）

请求示例

https://iam.api.ksyun.com/?Action=ListPolicies&公共请求参数

返回示例

- XML格式

```
<ListPoliciesResponse>
  <ListPoliciesResult>
    <Marker/>
    <IsTruncated>>false</IsTruncated>
    <Policies>
      <member>
        <PolicyId>4Ym5BEkcT4awlk30dIIQ_A</PolicyId>
        <DefaultVersionId>v1</DefaultVersionId>
        <UpdateDate>2016-05-16T17:30:43.000Z</UpdateDate>
        <AttachmentCount>0</AttachmentCount>
        <Description/>
        <Path/></Path>
        <CreateDate>2016-05-16T17:30:43.000Z</CreateDate>
        <Krn>krn:ksc:iam::73407420:policy/IAMadmin</Krn>
        <PolicyName>IAMadmin</PolicyName>
      </member>
      <member>
        <PolicyId>eCuWLaNkTp6Pjv3j348tNw</PolicyId>
        <DefaultVersionId>v1</DefaultVersionId>
        <UpdateDate>2016-05-16T17:43:32.000Z</UpdateDate>
        <AttachmentCount>0</AttachmentCount>
        <Description/>
        <Path/></Path>
        <CreateDate>2016-05-16T17:43:32.000Z</CreateDate>
        <Krn>krn:ksc:iam::73407420:policy/IAMadmin1</Krn>
        <PolicyName>IAMadmin1</PolicyName>
      </member>
    </Policies>
  </ListPoliciesResult>
  <ResponseMetadata>
    <RequestId>bf3c3c28-0576-4e9d-a8d8-bff28f75aeea</RequestId>
  </ResponseMetadata>
</ListPoliciesResponse>
```

- JSON示例

```
{
  "ListPoliciesResult": {
    "Policies": {
      "member": [
        {
          "PolicyId": "4Ym5BEkcT4awlK30dIIQ_A",
          "DefaultVersionId": "v1",
          "UpdateDate": "2016-05-16T17:30:43.000Z",
          "AttachmentCount": "0",
          "Path": "/",
          "CreateDate": "2016-05-16T17:30:43.000Z",
          "Krn": "krn:ksc:iam::73407420:policy/IAMAdmin",
          "PolicyName": "IAMAdmin"
        },
        {
          "PolicyId": "eCuWLaNkTp6Pjv3j348tNw",
          "DefaultVersionId": "v1",
          "UpdateDate": "2016-05-16T17:43:32.000Z",
          "AttachmentCount": "0",
          "Path": "/",
          "CreateDate": "2016-05-16T17:43:32.000Z",
          "Krn": "krn:ksc:iam::73407420:policy/IAMAdmin",
          "PolicyName": "IAMAdmin"
        }
      ]
    },
    "IsTruncated": false
  },
  "ResponseMetadata": {
    "RequestId": "caa91c07-c0ab-4ec0-9a30-556c657bfc60"
  }
}
```

新建策略版本内容

注1：当需要更新自定义策略的策略文档的内容时需要使用创建新策略版本的功能，任何自定义策略最多拥有5个策略版本；如果策略已经有5个策略版本，需要再次更新自定义策略时，则需要使用[DeletePolicyVersion](#)接口删除现存的某个策略版本。

注2：无法为系统策略和非属于自己的自定义策略创建新的策略版本。

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是	确定值	[a-zA-Z]+	操作接口名，系统规定参数，取值：CreatePolicyVersion
PolicyKrn	String	是	最短是25，最长2048	格式：krn:ksc:iam:: <i>account-id</i> : <i>policy/policy-name</i>	待创建新版本的策略的唯一标识
PolicyDocument	String	是	最短1，最长5K	[\w\s:~*{}[]/\$-]+	待创建新策略版本的策略文档内容
SetAsDefault	Boolean	否			可选参数，用于设置是否将新创建的策略版本指定为默认策略版本

返回元素（调用成功）

名称	类型	描述
PolicyVersion	PolicyVersionType	新创建策略版本的信息

错误码（调用不成功）

错误代码(Code)	错误消息(Message)	HTTP 状态码	中文描述（语义）
InvalidParameterValue	An value must be supplied for the input parameter %s.	400	输入参数%s的值不能为空，具体参数名在%s说明
InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter %s.	400	输入参数 %s的值无效、不合法或者超出范围，具体参数名在%s说明

PolicyVersionLimitExceeded	The request was rejected because it attempted to create policy version beyond the current KSC account limits.	409	请求被拒绝，因其创建的策略版本数量超过当前账户限制(PolicyVersion数量限制为5)
PolicyNoSuchEntity	The request was rejected because it referenced 'Policy' that does not exist.	404	请求被拒绝，因其引用的“策略”不存在
PolicyDocumentNotJsonFormat	The 'PolicyDocument' is not json format.	400	策略文档不是Json格式
PolicyDocumentInvalidVersion	The 'Version' in 'PolicyDocument' is invalid.	400	策略文档中的Version元素无效
PolicyDocumentInvalidStatement	The 'Statement' in 'PolicyDocument' is invalid.	400	策略文档中的Statement元素无效
PolicyDocumentInvalidEffect	The 'Effect' in 'PolicyDocument' is invalid.	400	策略文档中的Effect元素无效
PolicyDocumentInvalidResource	The 'Resource' in 'PolicyDocument' is invalid.	400	策略文档中的Resource元素无效
PolicyDocumentInvalidAction	The 'Action' in 'PolicyDocument' is invalid.	400	策略文档中的Action元素无效

示例（调用成功）

请求示例

```
https://iam.api.ksyun.com/?Action=CreatePolicyVersion
&PolicyKrn=krn:ksc:iam::12345678901234567890:policy/IAMAdmin
&PolicyDocument={"Version":"2015-11-01","Statement":[{"Effect":"Allow","Action":"iam:*",
"Resource":"*"}]}
&公共请求参数
```

返回示例

- XML格式

```
<CreatePolicyVersionResponse>
  <CreatePolicyVersionResult>
    <PolicyVersion>
      <IsDefaultVersion>>false</IsDefaultVersion>
      <VersionId>v2</VersionId>
      <CreateDate>2016-05-16T18:01:39.696Z</CreateDate>
    </PolicyVersion>
  </CreatePolicyVersionResult>
  <ResponseMetadata>
    <RequestId>253fc36f-e382-47ec-b44c-109553ea7d37</RequestId>
  </ResponseMetadata>
</CreatePolicyVersionResponse>
```

- JSON示例

```
{
  "CreatePolicyVersionResult": {
    "PolicyVersion": {
      "IsDefaultVersion": "false",
      "VersionId": "v2",
      "CreateDate": "2016-05-16T18:01:39.696Z"
    }
  },
  "ResponseMetadata": {
    "RequestId": "3e3dde15-327e-4289-804a-95683f143233"
  }
}
```

删除策略版本内容

注1：金山云账户只能删除自定义策略的策略版本内容，不能够删除系统策略的策略版本内容；且只能删除非默认策略版本，默认策略版本在调用[DeletePolicy](#)接口删除策略时被一并删掉。

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是	确定值	[a-zA-Z]+	操作接口名, 系统规定参数, 取值: GetPolicyVersion
PolicyKrn	String	是	最短是25, 最长2048	格式: krn:ksc:iam:: <i>account-id</i> : <i>policy/policy-name</i>	待删除策略版本的策略的唯一标识
VersionId	String	是	最短2	v[1-9][0-9]*	待删除的策略版本Id

返回元素 (调用成功)

无

错误码 (调用不成功)

错误代码 (Code)	错误消息 (Message)	HTTP 状态码	中文描述 (语义)
InvalidParameterValue	An value must be supplied for the input parameter %s.	400	输入参数%s的值不能为空, 具体参数名在%s说明
InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter %s.	400	输入参数 %s的值无效、不合法或者超出范围, 具体参数名在%s说明
PolicyNoSuchEntity	The request was rejected because it referenced 'Policy' that does not exist.	404	请求被拒绝, 因其引用的“策略”不存在
PolicyVersionNoSuchEntity	The request was rejected because it referenced 'PolicyVersion' that does not exist.	404	请求被拒绝, 因其引用的“策略版本”不存在
PolicyAccessDenied	The request was rejected because it attempted to attach a 'Policy' that you have no permission.	409	不能删除非属于本账户的自定义策略版本
PolicyDefaultVersionDeleteConflict	The request was rejected because it attempted to delete a 'PolicyVersion' that is the default version of the 'Policy'.	409	不能删除默认策略版本

示例 (调用成功)

请求示例

```
https://iam.api.ksyun.com/?Action=DeletePolicyVersion
&PolicyKrn=krn:ksc:iam::12345678901234567890:policy/IAMAdmin
&VersionId=v2
&公共请求参数
```

返回示例

- XML格式

```
<DeletePolicyVersionResponse>
  <ResponseMetadata>
    <RequestId>c00e9637-dfe8-44f1-9ac9-2df57b48f0c2</RequestId>
  </ResponseMetadata>
</DeletePolicyVersionResponse>
```

- JSON示例

```
{
  "ResponseMetadata": {
    "RequestId": "ae246270-cab5-4b48-8c0b-1645b7f6d0ad"
  }
}
```

查询策略版本内容

注1: 金山云账户能够查询自定义策略和系统策略的策略版本内容。

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
----	----	----	----------	------	----

Action	String	是	确定值	[a-zA-Z]+	操作接口名，系统规定参数，取值： GetPolicyVersion
PolicyKrn	String	是	最短是25，最 长2048	格式：krn:ksc:iam::account-id:po licy/policy-name	待查询的策略的唯一标识
VersionId	String	是	最短2	v[1-9][0-9]*	待查询的策略版本Id

返回元素（调用成功）

名称	类型	描述
PolicyVersion	PolicyVersionType	查询的策略版本的信息

错误码（调用不成功）

错误代码 (Code)	错误消息 (Message)	HTTP 状态码	中文描述（语义）
InvalidParameterValue	An value must be supplied for the input parameter %s.	400	输入参数%s的值不能为空，具体参数名在%s说明
InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter %s.	400	输入参数 %s的值无效、不合法或者超出范围，具体参数名在%s说明
PolicyNoSuchEntity	The request was rejected because it referenced 'Policy' that does not exist.	404	请求被拒绝，因其引用的“策略”不存在

示例（调用成功）

请求示例

```
https://iam.api.ksyun.com/?Action=GetPolicyVersion
&PolicyKrn=krn:ksc:iam::12345678901234567890:policy/IAMAdmin
&VersionId=v1
&公共请求参数
```

返回示例

- XML格式

```
<GetPolicyVersionResponse>
  <GetPolicyVersionResult>
    <PolicyVersion>
      <IsDefaultVersion>true</IsDefaultVersion>
      <VersionId>v1</VersionId>
      <CreateDate>2016-05-16T17:43:32.000Z</CreateDate>
      <Document>{"Version":"2015-11-01","Statement":[{"Effect":"Allow","Action":"iam:*","Resource":"*"}]}</Document>
    </PolicyVersion>
  </GetPolicyVersionResult>
  <ResponseMetadata>
    <RequestId>4c8e4c8a-ab39-43c5-86cf-553f9b97bc69</RequestId>
  </ResponseMetadata>
</GetPolicyVersionResponse>
```

- JSON示例

```
{
  "GetPolicyVersionResult": {
    "PolicyVersion": {
      "IsDefaultVersion": "true",
      "VersionId": "v1",
      "Document": "{\"Version\":\"2015-11-01\",\"Statement\":[{\"Effect\":\"Allow\",\"Action\":\"iam:*\", \"Resource\":\"*\"}]}"
    },
    "CreateDate": "2016-05-16T17:43:32.000Z"
  },
  "ResponseMetadata": {
    "RequestId": "6ba774b0-819c-4d2c-ad7a-10d70625e9bf"
  }
}
```

查询策略版本列表

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是	确定值	[a-zA-Z]+	操作接口名, 系统规定参数, 取值: ListPolicies
PolicyKrn	String	是	最短是25, 最长2048	格式: krn:ksc:iam::account-id:policy/policy-name	待查询策略版本列表的策略的唯一标识

返回元素 (调用成功)

名称	类型	描述
Versions	PolicyVersionType List	被查询的策略版本信息列表

错误码 (调用不成功)

错误代码 (Code)	错误消息 (Message)	HTTP 状态码	中文描述 (语义)
InvalidParameterValue	An value must be supplied for the input parameter %s.	400	输入参数%s的值不能为空, 具体参数名在%s说明
InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter %s.	400	输入参数 %s的值无效、不合法或者超出范围, 具体参数名在%s说明
PolicyNoSuchEntity	The request was rejected because it referenced 'Policy' that does not exist.	404	请求被拒绝, 因其引用的“策略”不存在

示例 (调用成功)

请求示例

```
https://iam.api.ksyun.com/?Action=ListPolicyVersions
&PolicyKrn=krn:ksc:iam::12345678901234567890:policy/IAMadmin
&公共请求参数
```

返回示例

- XML格式

```
<ListPolicyVersionsResponse>
  <ListPolicyVersionsResult>
    <Versions>
      <member>
        <IsDefaultVersion>false</IsDefaultVersion>
        <VersionId>v2</VersionId>
        <CreateDate>2016-05-16T18:01:39.000Z</CreateDate>
      </member>
      <member>
        <IsDefaultVersion>true</IsDefaultVersion>
        <VersionId>v1</VersionId>
        <CreateDate>2016-05-16T17:43:32.000Z</CreateDate>
      </member>
    </Versions>
  </ListPolicyVersionsResult>
  <ResponseMetadata>
    <RequestId>14961367-3e42-4e6d-a71f-d85c3cc80264</RequestId>
  </ResponseMetadata>
</ListPolicyVersionsResponse>
```

- JSON示例

```
{
  "ListPolicyVersionsResult": {
    "Versions": {
      "member": [
        {
          "IsDefaultVersion": "false",
          "VersionId": "v2",
          "CreateDate": "2016-05-16T18:01:39.000Z"
        },
        {
          "IsDefaultVersion": "true",
          "VersionId": "v1",
          "CreateDate": "2016-05-16T17:43:32.000Z"
        }
      ]
    }
  }
}
```

```

    }
  },
  "ResponseMetadata": {
    "RequestId": "2596f6c2-2263-4e17-94a1-1d0993d3aa3f"
  }
}

```

设定默认策略版本

注1: 修改自定义策略的默认策略版本会对所有附加该策略的实体（IAM本期只有“用户”一种实体）产生影响，可通过[ListEntitiesForPolicy](#)接口查询一个策略的附加实体。

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是	确定值	[a-zA-Z]+	操作接口名，系统规定参数，取值：SetDefaultPolicyVersion
PolicyKrn	String	是	最短是25，最长2048	格式：krn:ksc:iam::account-id:policy/policy-name	待设定默认策略版本的策略标识
VersionId	String	是	最短2	v[1-9][0-9]*	待设定为默认版本的策略版本Id

返回元素（调用成功）

无

错误码（调用不成功）

错误代码(Code)	错误消息(Message)	HTTP 状态码	中文描述（语义）
InvalidParameterValue	An value must be supplied for the input parameter %s.	400	输入参数%s的值不能为空，具体参数名在%s说明
InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter %s.	400	输入参数 %s的值无效、不合法或者超出范围，具体参数名在%s说明
PolicyNoSuchEntity	The request was rejected because it referenced 'Policy' that does not exist.	404	请求被拒绝，因其引用的“策略”不存在
PolicyVersionNoSuchEntity	The request was rejected because it referenced 'PolicyVersion' that does not exist.	404	请求被拒绝，因其引用的“策略版本”不存在

示例（调用成功）

请求示例

```

https://iam.api.ky.cn/?Action=SetDefaultPolicyVersion
&PolicyKrn=krn:ksc:iam::12345678901234567890:policy/IAMAdmin
&VersionId=v2
&公共请求参数

```

返回示例

- XML格式

```

<SetDefaultPolicyVersionResponse>
  <ResponseMetadata>
    <RequestId>ea6a0dbc-c26d-4085-87ad-45118564ada2</RequestId>
  </ResponseMetadata>
</SetDefaultPolicyVersionResponse>

```

- JSON示例

```

{
  "ResponseMetadata": {
    "RequestId": "0d4a742d-76ec-40d5-b5b8-839ed515a833"
  }
}

```

附加用户策略

注1：可以为用户附加全局系统策略或自定义策略。

注2：金山云的IAM只支持托管策略（managed policy），不支持内嵌策略（inline policy）。

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是	确定值	[a-zA-Z]+	操作接口名，系统规定参数，取值：AttachUserPolicy
PolicyKrn	String	是	最短是25，最长2048	格式：krn:ksc:iam::account-id:policy/policy-name	待附加的策略标识
UserName	String	是	最短1，最长64	[\w+=,.\@-]+	策略的附加目标用户名

返回元素（调用成功）

无

错误码（调用不成功）

错误代码(Code)	错误消息(Message)	HTTP 状态码	中文描述（语义）
InvalidParameterValue	An value must be supplied for the input parameter %s.	400	输入参数%s的值不能为空，具体参数名在%s说明
InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter %s.	400	输入参数 %s的值无效、不合法或者超出范围，具体参数名在%s说明
PolicyNoSuchEntity	The request was rejected because it referenced 'Policy' that does not exist.	404	请求被拒绝，因其引用的“策略”不存在
UserNoSuchEntity	The request was rejected because it referenced 'User' that does not exist.	404	请求被拒绝，因其引用的“用户”不存在
UserPolicyLimitExceeded	The request was rejected because it attempted to attach policy to the user that beyond the current KSC account limits.	409	请求被拒绝，因其欲为用户附加的策略数量超过当前账户限制(用户被附加的策略数量限制为5)
UserPolicyAlreadyExists	The request was rejected because it attempted to attach a policy to user that the policy already attached to the user	409	请求被拒绝，因用户已经附加过该策略

示例（调用成功）

请求示例

```
https://iam.api.ksyun.com/?Action=AttachUserPolicy
&PolicyKrn=krn:ksc:iam::12345678901234567890:policy/IAMAdmin
&UserName=Bob
&公共请求参数
```

返回示例

- XML格式

```
<AttachUserPolicyResponse>
  <ResponseMetadata>
    <RequestId>560a7123-108e-4fdf-b193-70fdd68147f9</RequestId>
  </ResponseMetadata>
</AttachUserPolicyResponse>
```

- JSON示例

```
{
  "ResponseMetadata": {
    "RequestId": "71d21ab4-c380-427b-bab8-e05e4838f7c1"
  }
}
```

分离用户策略

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是	确定值	[a-zA-Z]+	操作接口名，系统规定参数，取值：DetachUserPolicy
PolicyKrn	String	是	最短是25，最长2048	格式：krn:ksc:iam:: <i>account-id</i> :policy/ <i>policy-name</i>	待分离的策略标识
UserName	String	是	最短1，最长64	[\w+=,.\@-]+	策略的分离目标用户名

返回元素（调用成功）

无

错误码（调用不成功）

错误代码(Code)	错误消息(Message)	HTTP 状态码	中文描述（语义）
InvalidParameterValue	An value must be supplied for the input parameter %s.	400	输入参数%s的值不能为空，具体参数名在%s说明
InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter %s.	400	输入参数 %s的值无效、不合法或者超出范围，具体参数名在%s说明
PolicyNoSuchEntity	The request was rejected because it referenced 'Policy' that does not exist.	404	请求被拒绝，因其引用的“策略”不存在
UserNoSuchEntity	The request was rejected because it referenced 'User' that does not exist.	404	请求被拒绝，因其引用的“用户”不存在
UserPolicyNotAttached	The request was rejected because the user does not attach the policy.	404	请求被拒绝，因用户没有该策略

示例（调用成功）

请求示例

```
https://iam.api.ksc.com/?Action=DetachUserPolicy
&PolicyKrn=krn:ksc:iam::12345678901234567890:policy/IAMAdmin
&UserName=Bob
&公共请求参数
```

返回示例

- XML格式

```
<DetachUserPolicyResponse>
  <ResponseMetadata>
    <RequestId>72156b58-b22f-4d07-86a6-bdb0219f5e4b</RequestId>
  </ResponseMetadata>
</DetachUserPolicyResponse>
```

- JSON示例

```
{
  "ResponseMetadata": {
    "RequestId": "f8008e55-ad42-4c6c-bdfd-33a4e1368daf"
  }
}
```

查询用户附加策略列表

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是	确定值	[a-zA-Z]+	操作接口名, 系统规定参数, 取值: ListAttachedUserPolicies
Marker	String	否	最短1, 最长320	[:graph:]+	当返回结果被截断时, 使用本次返回的Marker值用于标示下次调用查询策略信息列表操作的起始点
MaxItems	Integer	否	最小值1, 最大值1000, 默认100		用于限制本次查询结果返回的策略数量, 如果仍有额外策略未显示, 则返回元素中的IsTruncated的值会被设置为true, 同时返回下次查询的起始点Marker值
PathPrefix	String	否	最短1, 最长512	/	可选参数, 用于过滤策略列表, 如果未传入则默认值是/
UserName	String	是	最短1, 最长64	[\w+,.@-]+	待查询策略列表的目标用户名

返回元素 (调用成功)

名称	类型	描述
AttachedPolicies	AttachedPolicyType List	被查询的附加策略信息列表
IsTruncated	Boolean	截断标志位, 用于说明是否仍有更多的策略信息未显示全
Marker	String	如果截断标志位是true, Marker值被设定, 用于标识下次查询的起始点

错误码 (调用不成功)

错误代码 (Code)	错误消息 (Message)	HTTP 状态码	中文描述 (语义)
InvalidParameterValue	An value must be supplied for the input parameter %s.	400	输入参数%s的值不能为空, 具体参数名在%s说明
InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter %s.	400	输入参数 %s的值无效、不合法或者超出范围, 具体参数名在%s说明
UserNoSuchEntity	The request was rejected because it referenced 'User' that does not exist.	404	请求被拒绝, 因其引用的“用户”不存在

示例 (调用成功)

请求示例

https://iam.api.ksyun.com/?Action=ListAttachedUserPolicies&UserName=Bob&公共请求参数

返回示例

- XML格式

```
<ListAttachedUserPoliciesResponse>
  <ListAttachedUserPoliciesResult>
    <Marker/>
    <IsTruncated>>false</IsTruncated>
    <AttachedPolicies>
      <member>
        <PolicyKrn>krn:ksc:iam::73407420:policy/IAMAdmin</PolicyKrn>
        <PolicyName>IAMAdmin</PolicyName>
      </member>
    </AttachedPolicies>
  </ListAttachedUserPoliciesResult>
  <ResponseMetadata>
    <RequestId>4dafb9b8-505a-407a-a37d-9cd40aeb3be9</RequestId>
  </ResponseMetadata>
</ListAttachedUserPoliciesResponse>
```

- JSON示例

```
{
  "ListAttachedUserPoliciesResult": {
    "IsTruncated": false,
    "AttachedPolicies": {
      "member": [
        {
          "PolicyKrn": "krn:ksc:iam::73407420:policy/IAMAdmin",
          "PolicyName": "IAMAdmin"
        }
      ]
    }
  }
}
```

```

    "PolicyKrn": "krn:ksc:iam::73407420:policy/IAMAdmin1",
    "PolicyName": "IAMAdmin1"
  }
]
},
"ResponseMetadata": {
  "RequestId": "b1bf26e1-fb9c-4783-9b94-baed82d043b7"
}
}

```

查询策略附加实体列表

注1：由于列表可能很长，因此可使用MaxItems和Marker来进行分页截断显示。

注2：金山云IAM一期目前支持IAM用户一种关联实体，因此结果只返回一种关联实体类型。

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是	确定值	[a-zA-Z]+	操作接口名，系统规定参数，取值：ListEntitiesForPolicy
PathPrefix	String	否	最短1, 最长512	√[:graph:]*	可选参数，用于过滤关联实体列表（目前只有用户一种类型的实体），如果未传入则默认值是/
Marker	String	否	最短1, 最长320	[:graph:]+	当返回结果被截断时，使用本次返回的Marker值用于标示下次调用查询关联实体操作的起始点
MaxItems	Integer	否	最小值1, 最大值1000, 默认100		用于限制本次查询结果返回的关联实体数量，如果仍有额外关联实体未显示，则返回元素中的IsTruncated的值会被设置为true，同时返回下次查询的起始点Marker值
PolicyKrn	String	是	最短是25, 最长2048	格式：krn:ksc:iam::account-id:policy/policy-name	待查询关联实体的策略标识

返回元素（调用成功）

名称	类型	描述
IsTruncated	Boolean	截断标志位，用于说明是否仍有更多的关联实体未显示全
Marker	String	如果截断标志位是true，Marker值被设定，用于标识下次查询的起始点
PolicyUsers	PolicyUserType List	策略关联用户信息列表

错误码（调用不成功）

错误代码 (Code)	错误消息 (Message)	HTTP 状态码	中文描述（语义）
InvalidParameterValue	An value must be supplied for the input parameter %s.	400	输入参数%s的值不能为空，具体参数名在%s说明
InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter %s.	400	输入参数 %s的值无效、不合法或者超出范围，具体参数名在%s说明
PolicyNoSuchEntity	The request was rejected because it referenced 'Policy' that does not exist.	404	请求被拒绝，因其引用的“策略”不存在

示例（调用成功）

请求示例

```

https://iam.api.ksyun.com/?Action=ListEntitiesForPolicy
&PolicyKrn=krn:ksc:iam::12345678901234567890:policy/IAMAdmin
&公共请求参数

```

返回示例

- XML格式

```

<ListEntitiesForPolicyResponse>
  <ListEntitiesForPolicyResult>

```



```

<Marker/>
<IsTruncated>>false</IsTruncated>
<PolicyUsers>
  <member>
    <UserName>Bob</UserName>
  </member>
</PolicyUsers>
</ListEntitiesForPolicyResult>
<ResponseMetadata>
  <RequestId>a3d501b8-e7af-444f-ab11-94d0ef3370fe</RequestId>
</ResponseMetadata>
</ListEntitiesForPolicyResponse>

```

- JSON示例

```

{
  "ListEntitiesForPolicyResult": {
    "PolicyUsers": {
      "member": [
        {
          "UserName": "Bob"
        }
      ]
    },
    "IsTruncated": false
  },
  "ResponseMetadata": {
    "RequestId": "66073ca3-c076-4dc2-aad1-9b800925a3d1"
  }
}

```

添加用户到用户组

请求参数

名称	类型	必填	长度限制	格式	描述
GroupName	string	是	1-128	[\w+,.@-]+	用户组名称，在一个账号下唯一
UserName	string	是	1-128	[\w+,.@-]+	账号下存在用户的用户名

返回元素

错误码

错误码	错误消息	Http Status Code
GroupNameInvalidParameterValue	An invalid or out-of-range value was supplied for the 'GroupName' parameter.	400
UsernameInvalidParameterValue	An invalid or out-of-range value was supplied for the 'Username' parameter.	400
GroupNoSuchEntity	The request was rejected because it referenced 'Group' that does not exist. The error message describes the entity.	404
UserNoSuchEntity	The request was rejected because it referenced 'User' that does not exist.	404
UserGroupRelationAlreadyExists	The request was rejected because it attempted to add user to a group that had joined.	409
ServiceFailure	The request processing has failed because of an unknown error, exception or failure.	500

示例

请求示例

```
http://10.146.16.202:8008/?Action=UpdateGroup&Version=2015-11-01&GroupName=admin&UserName=iam
```

返回示例

xml格式

```

<AddUserToGroupResponse>
  <result>1</result>

```

```
<ResponseMetadata>
  <RequestId>61f472fc-ea7e-4e89-9124-4f6e46116842</RequestId>
</ResponseMetadata>
</AddUserToGroupResponse>
```

json格式

附加策略到用户组

请求参数

名称	类型	必填	长度限制	格式	描述
GroupName	string	是	1-128	[\w+,.@-]+	用户组名称，在一个账号下唯一
PolicyKrn	string	是	20-2048	krn:ksc:iam::account-id:policy/policy-name	策略Krn，系统策略或者是账户的策略

返回元素

错误码

错误码	错误信息	HTTP Status Code
PolicyKrnInvalidParameterValue	An invalid or out-of-range value was supplied for the 'PolicyKrn' parameter .	400
GroupNameInvalidParameterValue	An invalid or out-of-range value was supplied for the 'GroupName' parameter .	400
PolicyAccessDenied	The request was rejected because it attempted to attach a 'Policy' that you have no permission.	409
PolicyNoSuchEntity	The request was rejected because it referenced 'Policy' that does not exist. The error message describes the entity.	404
GroupNoSuchEntity	The request was rejected because it referenced 'Group' that does not exist. The error message describes the entity.	404
GroupPolicyAlreadyExists	The request was rejected because it attempted to create a policy that already exists.	409
ServiceFailure	The request processing has failed because of an unknown error, exception or failure.	500

示例

请求示例

```
http://10.146.16.202:8008/?Action=AttachGroupPolicy&Version=2015-11-01&GroupName=admin&PolicyKrn=krn:ksc:iam::ksc:policy/AdministratorAccess
```

返回示例

xml格式

```
<AttachGroupPolicyResponse>
  <result>1774</result>
  <ResponseMetadata>
    <RequestId>c2c217a7-71f8-4a78-be57-f8608a1205d7</RequestId>
  </ResponseMetadata>
</AttachGroupPolicyResponse>
```

json格式

创建用户组

创建用户组(CreateGroup)

描述

请求参数

名称	类型	必填	长度限制	格式	描述
GroupName	string	是	1-128	[\w+=,.\@-]+	用户组名称
Description	string	否	1-128		用户组描述

返回元素

错误码

错误码	错误信息	HTTP Status Code
GroupNameInvalidParameterValue	An invalid or out-of-range value was supplied for the 'GroupName' parameter.	400
GroupDescInvalidParameterValue	An invalid or out-of-range value was supplied for the 'Description' parameter.	400
GroupLimitExceeded	The request was rejected because it attempted to create user group beyond the current KSC account limits. The error message describes the limit exceeded.	409
GroupAlreadyExists	The request was rejected because it attempted to create a group that already exists.	409
ServiceFailure	The request processing has failed because of an unknown error, exception or failure.	500

示例

请求示例

http://10.146.16.202:8008/?Action=CreateGroup&Version=2015-11-01&GroupName=admin&Description=管理组

返回示例

xml格式

```
<CreateGroupResponse>
  <CreateGroupResult>
    <Group>
      <GroupId>KshVAnKrT821JsaCgcFIsg</GroupId>
      <Path></Path>
      <GroupName>console</GroupName>
      <Description>控制台组</Description>
      <CreateDate>2017-10-31T09:34:22Z</CreateDate>
      <Krn>krn:ksc:iam::2000000719:group/console</Krn>
    </Group>
  </CreateGroupResult>
  <ResponseMetadata>
    <RequestId>b3ef2d87-5599-4487-ab7a-317dc81fe5f0</RequestId>
  </ResponseMetadata>
</CreateGroupResponse>
```

json格式

```
{
  "RequestId": "269b29cf-f692-470a-bb3d-ec06cbd42e53",
  "Error": {
    "Type": "sender",
    "Code": "GroupAlreadyExists",
    "Message": "The request was rejected because it attempted to create a group that already exists."
  }
}
```

删除用户组

请求参数

名称	类型	必填	长度限制	格式	描述
GroupName	string	是	1-128	[\w+=,.\@-]+	用户组名

错误码

错误码	错误信息	HTTP Status Code
GroupNameInvalidParameterValue	An invalid or out-of-range value was supplied for the 'GroupName' parameter.	400
GroupNoSuchEntity	The request was rejected because it referenced 'Group' that does not exist. The error message describes the entity.	409
GroupUserDeleteConflict	The request was rejected because it attempted to delete a group that has users. Delete users belong to group before delete the group.	409
GroupPolicyDeleteConflict	The request was rejected because it attempted to delete a group that has policies. Detach policies belong to group before delete the group.	409
ServiceFailure	The request processing has failed because of an unknown error, exception or failure.	500

示例

请求示例

http://10.146.16.202:8008/?Action=DeleteGroup&Version=2015-11-01&GroupName=admin

返回示例

xml格式

```
<DeleteGroupResponse>
  <result>1</result>
  <ResponseMetadata>
    <RequestId>e1357bda-1cde-4321-8a4d-cf6a40b0e409</RequestId>
  </ResponseMetadata>
</DeleteGroupResponse>
```

json格式

从用户组分离策略

请求参数

名称	类型	必填	长度限制	格式	描述
GroupName	string	是	1-128	[\w+,.@-]+	用户组名
PolicyKrn	string	是	20-2048	krn:ksc:iam::12345678:policy/policy-name	权限的krn, 系统权限或用户自身权限

返回元素

错误码

错误码	错误信息	HTTP Status Code
InvalidInput	The request was rejected because an invalid or out-of-range value was supplied for an input parameter.	400
LimitExceeded	The request was rejected because it attempted to create resources beyond the current KSC account limits. The error message describes the limit exceeded.	409
NoSuchEntity	The request was rejected because it referenced an entity that does not exist. The error message describes the entity.	404
ServiceFailure	The request processing has failed because of an unknown error, exception or failure.	500

示例

请求示例

http://10.146.16.202:8008/?Action=DetachGroupPolicy&Version=2015-11-01&GroupName=admin&PolicyKrn=krn:ksc:iam::ksc:policy/AdministratorAccess

返回示例

xml 格式

```
<DetachGroupPolicyResponse>
  <result>1</result>
  <ResponseMetadata>
    <RequestId>05d719ab-6c70-4cc8-88c2-1724381ffaea</RequestId>
  </ResponseMetadata>
</DetachGroupPolicyResponse>
```

json 格式

获取用户组

请求参数

名称	类型	必填	长度限制	格式	描述
GroupName	string	是	1-128	[\w+=. , @-]+	用户组名称
Marker	string	否	1-320	[\u0020-\u00FF]+	Marker
MaxItems	integer	否	1-1000	integer	默认值100

返回元素

错误码

错误码	错误信息	HTTP Status Code
GroupNameInvalidParameterValue	An invalid or out-of-range value was supplied for the 'GroupName' parameter.	400
GroupNoSuchEntity	The request was rejected because it referenced 'Group' that does not exist. The error message describes the entity.	409
ServiceFailure	The request processing has failed because of an unknown error, exception or failure.	500

示例

请求示例

返回示例

xml 格式

```
<GetGroupResponse>
  <GetGroupResult>
    <Group>
      <Group>
        <GroupId>KV1mh19FRhKTCbxBZoa5Fw</GroupId>
        <Path></Path>
        <GroupName>admin</GroupName>
        <Description>管理组</Description>
        <CreateDate>2017-11-06T10:01:19Z</CreateDate>
        <Krn>krn:ksc:iam::2000000719:group/admin</Krn>
      </Group>
    </Group>
    <Users>
      <Item>
        <UserId>v5saAmquTumE6iLhU5yIoQ</UserId>
        <Path></Path>
        <UserName>iam</UserName>
        <RealName>测试</RealName>
        <CreateDate>2017-11-06T06:21:43Z</CreateDate>
        <Phone>18612708815</Phone>
        <Email>rjlz@163.com</Email>
        <PhoneVerified>No</PhoneVerified>
        <EmailVerified>No</EmailVerified>
        <Remark></Remark>
        <Krn>krn:ksc:iam::2000000719:user/iam</Krn>
        <PasswordResetRequired>1</PasswordResetRequired>
      </Item>
    </Users>
    <IsTruncated></IsTruncated>
  </GetGroupResult>
</GetGroupResponse>
```

```

    <ResponseMetadata>
      <RequestId>ffb1cf4e-db6d-4d1c-b609-4ba8c4828ccf</RequestId>
    </ResponseMetadata>
  </ListGroupResponse>

```

json格式

获取用户组权限列表

获取用户组权限列表(ListGroupPolicies)

请求参数

名称	类型	必填	长度限制	格式	描述
GroupName	string	是	1-128	[\w+=,.\@-]+	用户组名称
Marker	string	否	1-320	[\u0020-\u00FF]+	Marker
MaxItems	string	否	1-1000	-	最大返回个数，默认值100

返回元素

错误码

错误码	错误信息	HTTP Status Code
GroupNameInvalidParameterValue	An invalid or out-of-range value was supplied for the 'GroupName' parameter.	400
GroupNoSuchEntity	The request was rejected because it referenced 'Group' that does not exist. The error message describes the entity.	409
ServiceFailure	The request processing has failed because of an unknown error, exception or failure.	500

示例

请求示例

返回示例

xml格式

```

<ListGroupPoliciesResponse>
  <ListAttachedRolePoliciesResult>
    <AttachedPolicies>
      <member>
        <Item>
          <PolicyKrn>krn:ksc:iam::ksc:policy/AdministratorAccess</PolicyKrn>
          <PolicyName>AdministratorAccess</PolicyName>
        </Item>
      </member>
    </AttachedPolicies>
    <IsTruncated></IsTruncated>
  </ListAttachedRolePoliciesResult>
  <ResponseMetadata>
    <RequestId>e75a9bcc-3eee-492a-bf9e-81d06ad8b2ab</RequestId>
  </ResponseMetadata>
</ListGroupPoliciesResponse>

```

json格式

获取用户组列表

获取用户组列表(ListGroups)

请求参数

名称	类型	必填	长度限制	格式	描述
Marker	string	否	1-320	[\u0020-\u00FF]+	Marker

MaxItems integer 否 1-1000 - 最大返回数目，默认100

返回元素

错误码

错误码	错误信息	HTTP Status Code
ServiceFailure	The request processing has failed because of an unknown error, exception or failure.	500

示例

请求示例

http://10.146.16.202:8008/?Action=ListGroup&Version=2015-11-01

返回示例

xml格式

```
<ListGroupResponse>
  <ListGroupResult>
    <Groups>
      <member>
        <Item>
          <GroupId>Ir047A4zT2myQDWAVUMztw</GroupId>
          <Path>/</Path>
          <GroupName>console</GroupName>
          <Description>控制台组</Description>
          <CreateDate>2017-11-02T08:40:41Z</CreateDate>
          <Krn>krn:ksc:iam::2000000719:group/console</Krn>
        </Item>
        <Item>
          <GroupId>hvoCKPL-TD0tyGzXQtz80w</GroupId>
          <Path>/</Path>
          <GroupName>admin</GroupName>
          <Description>管理组</Description>
          <CreateDate>2017-11-06T09:32:14Z</CreateDate>
          <Krn>krn:ksc:iam::2000000719:group/admin</Krn>
        </Item>
      </member>
    </Groups>
    <IsTruncated></IsTruncated>
  </ListGroupResult>
  <ResponseMetadata>
    <RequestId>1350777c-0bc4-431f-bf29-4efe1b1b92db</RequestId>
  </ResponseMetadata>
</ListGroupResponse>
```

json格式

获取用户所属用户组列表

获取用户所属用户组列表(ListGroupsForUser)

请求参数

名称	类型	必填	长度限制	格式	描述
Marker	string	否	1-320	[\u0020-\u00FF]	Marker
MaxItems	int	否	1-1000	-	最大返回数据数目
UserName	string	是	1-128	[\w+=[.@-]+	用户名

返回元素

错误码

错误码	错误信息	HTTP Status Code
-----	------	------------------

UsernameInvalidParameterValue	An invalid or out-of-range value was supplied for the 'Username' parameter.	400
UserNoSuchEntity	The request was rejected because it referenced 'User' that does not exist.	404
ServiceFailure	The request processing has failed because of an unknown error, exception or failure.	500

示例

请求示例

http://10.146.16.202:8008/?Action=ListGroupForUser&Version=2015-11-01&UserName=lvzhao

返回示例

xml格式

```
<ListGroupForUserResponse>
  <result>
    <Groups>
      <member>
        <Item>
          <GroupId>KV1mhi9FRhKTCbxBZoa5Fw</GroupId>
          <Path></Path>
          <GroupName>admin</GroupName>
          <Description>管理组</Description>
          <CreateDate>2017-11-06T10:01:19Z</CreateDate>
          <Krn>krn:ksc:iam::2000000719:group/admin</Krn>
        </Item>
      </member>
    </Groups>
    <IsTruncated></IsTruncated>
  </result>
  <ResponseMetadata>
    <RequestId>55ebb4c8-65b6-4594-a722-21c8f84c39c2</RequestId>
  </ResponseMetadata>
</ListGroupForUserResponse>
```

json格式

从用户组删除用户

从用户组删除用户 (RemoveUserFromGroup)

描述

请求参数

名称	类型	必填	长度限制	格式	描述
GroupName	string	是	1-128	[\w+,.@-]+	用户组名称
UserName	string	是	1-128	[\w+,.@-]+	用户名称

返回元素

错误码

错误码	错误信息	HTTP Status Code
GroupNameInvalidParameterValue	An invalid or out-of-range value was supplied for the 'GroupName' parameter.	400
UsernameInvalidParameterValue	An invalid or out-of-range value was supplied for the 'Username' parameter.	400
GroupNoSuchEntity	The request was rejected because it referenced 'Group' that does not exist. The error message describes the entity.	404
UserNoSuchEntity	The request was rejected because it referenced 'User' that does not exist.	404
UserGroupRelationNoSuchEntity	The request was rejected because the user does not belong to the group.	404

ServiceFailure The request processing has failed because of an unknown error, exception or failure. 500

示例

请求示例

http://10.146.16.202:8008/?Action=RemoveUserFromGroup&Version=2015-11-01&GroupName=admin&UserName=iam

返回示例

xml 格式

```
<RemoveUserFromGroupResponse>
  <result>1</result>
  <ResponseMetadata>
    <RequestId>302bac0c-ed89-4e02-b641-1ec4c9d8d953</RequestId>
  </ResponseMetadata>
</RemoveUserFromGroupResponse>
```

json 格式

更新用户组

更新用户组 (UpdateGroup)

请求参数

名称	类型	必填	长度限制	格式	描述
GroupName	string	是	1-128	[\w+=,.\@-]+	用户组名称
NewGroupName	string	是	1-128	[\w+=,.\@-]+	新的用户组

返回元素

错误码

错误码	错误信息	HTTP Status Code
GroupNameInvalidParameterValue	An invalid or out-of-range value was supplied for the 'GroupName' parameter.	400
GroupDescInvalidParameterValue	An invalid or out-of-range value was supplied for the 'Description' parameter.	400
GroupNoSuchEntity	The request was rejected because it referenced 'Group' that does not exist. The error message describes the entity.	409
ServiceFailure	The request processing has failed because of an unknown error, exception or failure.	500

示例

请求示例

http://10.146.16.202:8008/?Action=UpdateGroup&Version=2015-11-01&GroupName=console&NewGroupName=console1

返回示例

xml 格式

```
<UpdateGroupResponse>
  <result>1</result>
  <ResponseMetadata>
    <RequestId>4f9e9c25-51cb-4423-ad0c-c3a679f66f33</RequestId>
  </ResponseMetadata>
</UpdateGroupResponse>
```

json 格式

创建角色

概述

创建角色。金山云账号或被授权的子用户可以调用该接口。

创建Role需要提供信任账号列表,以逗号分隔。

注：一个账号下最多拥有100个Role

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是		CreateRole	action名称
RoleName	String	是	最短1, 最长64	[\w+=,.\@-]+	
TrustAccounts	String	是	最短1, 最长1000		信任账号列表(多个账号之间使用半角逗号(,)分隔, 最多可以有20个账号)
Path	String	否	最短:1, 最长:512	√ or \[:graph:]+√	默认是/, 该字段暂未支持, 一律处理为 '/'
Description	String	否	最短:1, 最长:128		默认是空字符串

返回元素 (调用成功)

名称	类型	描述
Role		新创建的角色基本信息

错误码 (调用不成功)

错误代码	描述	HTTP 状态码	语义
MissingParameter	A required parameter for the specified action is not supplied.	400	忘记输入参数, 具体参数名在错误消息中说明(RoleName)
InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter.	400	参数格式/取值范围不对, 具体参数名在错误消息中说明(RoleName)
LimitExceeded	The request was rejected because it attempted to create resources beyond the current Ksyun account limits	409	资源数量超过限制, 具体限制在错误消息中说明
EntityAlreadyExists	The request was rejected because it attempted to create a resource that already exists	409	资源已经存在, 具体实体名称在错误消息中说明

示例 (调用成功)

请求示例

```
https://iam.cn-beijing-6.api.ksyun.com/?Action=CreateRole
&RoleName=roleTest
&TrustAccounts=123123,456456,789679
&Version=2015-11-01
&AUTHPARAMS
```

返回示例

- XML示例

```
<CreateRoleResponse>
  <CreateRoleResult>
    <Role>
      <Path>/ksc/iam/</Path>
      <Krn>krn:kcs:iam::213:role/ksc/iam/roleTest</Krn>
      <RoleName>roleTest</RoleName>
      <Description>这个一个测试使用的Role</Description>
      <TrustAccounts>123123,456456,789679</TrustAccounts>
      <CreateDate>2017-02-08T23:34:01.495Z</CreateDate>
      <RoleId>AR0ADBQP57FF2AEXAMPLE</RoleId>
    </Role>
  </CreateRoleResult>
</CreateRoleResponse>
```

```
</CreateRoleResult>
<ResponseMetadata>
  <RequestId>4a93ceee-9966-11e1-b624-b1aEXAMPLE7c</RequestId>
</ResponseMetadata>
</CreateRoleResponse>
```

删除角色

金山云账号或被授权的子用户可以调用该接口。

约束：只有角色没有附加任何访问策略，才可以被删除。

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是		DeleteRole	action名称
RoleName	String	是	最短1, 最长64	[\w+=,.\@-]+	

返回元素（调用成功）

无

错误码（调用不成功）

错误代码	描述	HTTP 状态码	语义
MissingParameter	A required parameter for the specified action is not supplied.	400	忘记输入参数，具体参数名在错误消息中说明
InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter.	400	参数格式/取值范围不对，具体参数名在错误消息中说明
NoSuchEntity	The request was rejected because it referenced a resource that does not exist	404	实体不存在，具体实体名称在错误消息中说明

示例（调用成功）

请求示例

```
https://iam.cn-beijing-6.api.ksyun.com/?Action=DeleteRole
&RoleName=roleTest
&Version=2015-11-01
&AUTHPARAMS
```

返回示例

- XML示例

```
<DeleteRoleResponse>
  <ResponseMetadata>
    <RequestId>913e3f37-99ed-11e1-a4c3-270EXAMPLE04</RequestId>
  </ResponseMetadata>
</DeleteRoleResponse>
```

查询账号的角色列表

查询本账号内已创建的角色列表信息。金山云账号或被授权的子用户可以调用该接口。

如果PathPrefix参数为空，默认为/，列出所有的角色。

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是	ListRoles		action名称
Marker	String	否	最短1, 最长320	[:graph:]+	当返回结果被截断时，使用本次返回的Marker值用于标示下次调用查询操作的起始点，Marker值不会过期

MaxItems	Integer	否	最小值1, 最大值1000, 默认100	用于限制本次查询结果返回的数量, 如果仍有额外元素未显示, 则返回元素中的IsTruncated的值会被设置为true, 同时返回下次查询的起始点Marker值
PathPrefix	String	否	最短1, 最长512	可选参数, 用于过滤列表, 如果未传入则默认值是/(/[A-Za-z0-9\.\, \+@=_-]+)*)/

返回元素 (调用成功)

名称	类型	描述
IsTruncated	Boolean	截断标志位, 用于说明是否仍有更多的元素未显示全
Marker	String	如果截断标志位是true, Marker值被设定, 用于标识下次查询的起始点
Roles.member List		角色列表

错误码 (调用不成功)

错误代码	描述	HTTP 状态码	语义
InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter.	400	参数格式/取值范围不对, 具体参数名在错误消息中说明

示例 (调用成功)

请求示例

```
https://iam.cn-beijing-6.api.ksyun.com/?Action=ListRoles
&MaxItems=100
&Version=2015-11-01
&AUTHPARAMS
```

返回示例

- XML示例

```
<ListRolesResponse>
<ListRolesResult>
  <IsTruncated>>false</IsTruncated>
  <Roles>
    <member>
      <Path>/ksc/iam/</Path>
      <Krn>krn:kcs:iam::213:role/ksc/iam/roleTest</Krn>
      <RoleName>roleTest</RoleName>
      <Description>这个一个测试使用的Role</Description>
      <TrustAccounts>123123, 456456, 789679</TrustAccounts>
      <CreateDate>2017-02-08T23:34:01.495Z</CreateDate>
      <RoleId>AR0ADBQP57FF2AEXAMPLE</RoleId>
    </member>
  </Roles>
</ListRolesResult>
<ResponseMetadata>
  <RequestId>20f7279f-99ee-11e1-a4c3-27EXAMPLE804</RequestId>
</ResponseMetadata>
</ListRolesResponse>
```

查询角色基本信息

查询角色基本信息。金山云账号或被授权的子用户可以调用该接口。

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是		GetRole	action名称
RoleName	String	是	最短1, 最长64	[\w+=,.\@-]+	

返回元素 (调用成功)

名称	类型	描述
----	----	----

Role 新创建的角色基本信息

错误码（调用不成功）

错误代码	描述	HTTP 状态码	语义
InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter.	400	参数格式/取值范围不对，具体参数名在错误消息中说明
NoSuchEntity	The request was rejected because it referenced a resource that does not exist	404	实体不存在，具体实体名称在错误消息中说明

示例（调用成功）

请求示例

```
https://iam.cn-beijing-6.api.ksyun.com/?Action=GetRole
&RoleName=roleTest
&Version=2015-11-01
&AUTHPARAMS
```

返回示例

- XML示例

```
<GetRoleResponse>
  <GetRoleResult>
    <Role>
      <Path>/ksc/iam/</Path>
      <Krn>krn:kcs:iam::213:role/ksc/iam/roleTest</Krn>
      <RoleName>roleTest</RoleName>
      <Description>这个一个测试使用的Role</Description>
      <TrustAccounts>123123,456456,789679</TrustAccounts>
      <CreateDate>2017-02-08T23:34:01.495Z</CreateDate>
      <RoleId>AR0ADBQP57FF2AEXAMPLE</RoleId>
    </Role>
  </GetRoleResult>
  <ResponseMetadata>
    <RequestId>4a93ceee-9966-11e1-b624-b1aEXAMPLE7c</RequestId>
  </ResponseMetadata>
</GetRoleResponse>
```

更新角色的信任策略

概述

更新角色的信任策略。金山云账号或被授权的子用户可以调用该接口。

注：默认最多添加20个信任账号；该接口是替换全部信任账号（删除所有旧数据，再添加新数据）

openAPI接口

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是		UpdateRoleTrustAccounts	action名称
RoleName	String	是	最短1, 最长64	[\w+,.@-]+	
NewTrustAccounts	String	是	最短1, 最长1000		信任账号列表(多个账号之间使用半角逗号(,)分隔, 最多可以有20个账号)

返回元素（调用成功）

无

错误码（调用不成功）

错误代码	描述	HTTP 状态码	语义
InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter.	400	参数格式/取值范围不对，具体参数名在错误消息中说明
NoSuchEntity	The request was rejected because it referenced a resource that does not exist	404	实体不存在，具体实体名称在错误消息中说明

示例（调用成功）

请求示例

```
https://iam.cn-beijing-6.api.ksyun.com/?Action=UpdateRoleTrustAccounts
&NewTrustAccounts=123123,456456,789679
&RoleName=testRole
&Version=2015-11-01
&AUTHPARAMS
```

返回示例

- XML示例

```
<UpdateRoleTrustAccountsResponse>
  <ResponseMetadata>
    <RequestId>309c1671-99ed-11e1-a4c3-270EXAMPLE04</RequestId>
  </ResponseMetadata>
</UpdateRoleTrustAccountsResponse>
```

附加角色访问策略

概述

为Role附加访问策略。金山云账号或被授权的子用户可以调用该接口。附加的策略定义了Role的访问权限。

注：每个角色附加的访问策略总数最多10个

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是		AttachRolePolicy	action名称
RoleName	String	是	最短1, 最长64	[\w+=[,.\@-]+	
PolicyKrn	String	是	最短是25, 最长2048	格式: krn:ksc:iam:: <i>account-id</i> :policy/ <i>policy-name</i>	待附加的策略标识

返回元素（调用成功）

无

错误码（调用不成功）

错误代码	描述	HTTP 状态码	语义
InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter.	400	参数格式/取值范围不对，具体参数名在错误消息中说明 (UserName)
NoSuchEntity	The request was rejected because it referenced a resource that does not exist	404	实体不存在，具体实体名称在错误消息中说明
LimitExceeded	The request was rejected because it attempted to attach policy to the role that beyond the current KSC account limits. The error message describes the limit exceeded.	409	角色被附加的策略数量超过限制

示例（调用成功）

请求示例

```
https://iam.cn-beijing-6.api.ksyun.com/?Action=AttachRolePolicy
&PolicyKrn=krn:ksc:iam::ksc:policy/ReadOnlyAccess
&RoleName=ReadOnlyRole
&Version=2015-11-01
&AUTHPARAMS
```

返回示例

- XML示例

```
<AttachRolePolicyResponse>
  <ResponseMetadata>
    <RequestId>309c1671-99ed-11e1-a4c3-270EXAMPLE04</RequestId>
  </ResponseMetadata>
</AttachRolePolicyResponse>
```

分离角色访问策略

分离角色访问策略。金山云账号或被授权的子用户可以调用该接口。

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是		DetachRolePolicy	action名称
RoleName	String	是	最短1, 最长64	[\w+=, .@-]+	
PolicyKrn	String	是	最短是25, 最长2048	格式: krn:ksc:iam:: <i>account-id</i> :policy/ <i>policy-name</i>	待分离的策略标识

返回元素（调用成功）

无

错误码（调用不成功）

错误代码	描述	HTTP 状态码	语义
InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter.	400	参数格式/取值范围不对, 具体参数名在错误消息中说明(UserName)
NoSuchEntity	The request was rejected because it referenced a resource that does not exist	404	实体不存在, 具体实体名称在错误消息中说明

示例（调用成功）

请求示例

```
https://iam.cn-beijing-6.api.ksyun.com/?Action=DetachRolePolicy
&PolicyKrn=krn:ksc:iam::KSC:policy/ReadOnlyAccess
&RoleName=ReadOnlyRole
&Version=2015-11-01
&AUTHPARAMS
```

返回示例

- XML示例

```
<DetachRolePolicyResponse>
  <ResponseMetadata>
    <RequestId>309c1671-99ed-11e1-a4c3-270EXAMPLE04</RequestId>
  </ResponseMetadata>
</DetachRolePolicyResponse>
```

查询角色附加的策略列表

概述

查询角色附加的策略列表。金山云账号或被授权的子用户可以调用该接口。

注：如果附加策略列表很长，可使用MaxItems（结合Marker）来进行分页截断显示。

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是		ListAttachedRolePolicies	action名称
PathPrefix	String	否	最短1, 最长512	((/[A-Za-z0-9\\.,\\+@=-_]+)*)/	可选参数，用于过滤列表，如果未传入则默认值是/
Marker	String	否	最短1, 最长320	[[:graph:]]+	当返回结果被截断时，使用本次返回的Marker值用于标示下次调用查询操作的起始点，Marker值不会过期
MaxItems	Integer	否	最小值1, 最大值1000, 默认100		用于限制本次查询结果返回的数量，如果仍有额外元素未显示，则返回元素中的IsTruncated的值会被设置为true，同时返回下次查询的起始点Marker值
RoleName	String	是	最短1, 最长64	[\\w+=,\\.@-]+	

返回元素（调用成功）

名称	类型	描述
IsTruncated	Boolean	截断标志位，用于说明是否仍有更多的元素未显示全
Marker	String	如果截断标志位是true，Marker值被设定，用于标识下次查询的起始点
AttachedPolicies	List	附加策略列表

错误码（调用不成功）

错误代码	描述	HTTP 状态码	语义
InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter.	400	参数格式/取值范围不对，具体参数名在错误消息中说明
NoSuchEntity	The request was rejected because it referenced a resource that does not exist	404	实体不存在，具体实体名称在错误消息中说明

示例（调用成功）

请求示例

```
https://iam.cn-beijing-6.api.ksyun.com/?Action=ListAttachedRolePolicies
&RoleName=ReadOnlyRole
&Version=2015-11-01
&AUTHPARAMS
```

返回示例

- XML示例

```
<ListAttachedRolePoliciesResponse>
  <ListAttachedRolePoliciesResult>
    <AttachedPolicies>
      <member>
        <PolicyName>ReadOnlyAccess</PolicyName>
        <PolicyKrn>arn:ksc:iam:ksc:policy/ReadOnlyAccess</PolicyKrn>
      </member>
    </AttachedPolicies>
    <IsTruncated>>false</IsTruncated>
  </ListAttachedRolePoliciesResult>
  <ResponseMetadata>
    <RequestId>9a3b490d-3ea5-11e4-9d0d-6f969EXAMPLE</RequestId>
  </ResponseMetadata>
</ListAttachedRolePoliciesResponse>
```


获取角色的临时身份

概述

通过扮演角色，获取角色的临时身份 在扮演角色前，请确认是否有相应的权限：1. 访问者有调用AssumeRole的权限 2. 角色的信任账号列表中需要包含访问者所在账号。

注1：该接口只能以子用户或者角色的身份进行调用。在调用其他OpenAPI的时候，请在header头增加X-Ksc-Security-Token:SecurityToken（会话）。不需要再算签名时包含该header头

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是		AssumeRole	action名称
RoleKrn	String	是	min:20, max:2048	krn:ksc:iam::account-id:role/role-name	
RoleSessionName	String	是	min:2, max:64	[\w+,.@-]*	用户自定义参数,此参数区分不同的临时证书
DurationSeconds	String	否	min:900, max:3600, 默认 3600		指定临时证书的持续有效时间
Policy	String	否	min:1, max:2048	[\u0009\u000A\u000D\u0020-\u00FF]+	如果传递了policy，最终值将取policy和role的（这一期先不实现该功能）access_policy的交集。没有传递该参数，将获取role附加的access_policy的所有权限

返回元素（调用成功）

名称	类型	描述
AssumeRoleResult		AssumeRole的临时证书

错误码（调用不成功）

错误代码	描述	HTTP 状态码	语义
InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter.	400	参数格式/取值范围不对，具体参数名在错误消息中说明(UserName)
NoSuchEntity	The request was rejected because it referenced a resource that does not exist	404	实体不存在，具体实体名称在错误消息中说明
InvalidClientTokenId	The access key ID provided does not exist in our records.	403	无效的accessKey
AccessDenied	用户krn:ksc:iam::%s:user%s没有权限执行操作：%s:%s。	403	没有执行该操作的权限

示例（调用成功）

请求示例

```
http://sts.cn-beijing-6.api.ksyun.com/?Action=AssumeRole
&Version=2015-11-01
&RoleSessionName=Bob
&RoleKrn=krn:ksc:iam::123456789012:role/demo
&AUTHPARAMS
```

返回示例

- xml示例

```
<AssumeRoleResponse>
  <AssumeRoleResult>
    <Credentials>
      <SecretAccessKey>wJalrXUtnFEMI/K7MDENG/bPxRfiCYzEXAMPLEKEY</SecretAccessKey>
      <Expiration>2017-07-15T23:28:33.359Z</Expiration>
      <AccessKeyId>AKIAIOSFODNN7EXAMPLE</AccessKeyId>
```

```

    <SecurityToken>V1xxxxxxxxxx</SecurityToken>
  </Credentials>
  <AssumedRoleUser>
    <Krn>krn:ksc:sts::123456789012:assumed-role/demo/Bob</Krn>
    <AssumedRoleId>ARO123EXAMPLE123:Bob</AssumedRoleId>
  </AssumedRoleUser>
</AssumeRoleResult>
<ResponseMetadata>
  <RequestId>c6104cbe-af31-11e0-8154-cbc7ccf896c7</RequestId>
</ResponseMetadata>
</AssumeRoleResponse>

```

创建虚拟设备

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是	确定值	[a-zA-Z]+	操作接口名，系统规定参数，取值：CreateVirtualMFADevice
Path	String	否	最短1，最长512	√ or \[:graph:]+√	可选参数，用户名的路径，如果未传入则默认值是/
VirtualMFADeviceName	String	是	最短1，最长64	^[\\w\\Q+,.@-\\E]+\$	虚拟MFA的名字

返回元素（调用成功）

名称	类型	描述
VirtualMFADevice	VirtualMFADevice	虚拟设备

错误码（调用不成功） {#错误码（调用不成功）}

错误代码	描述	HTTP 状态码	语义
MissingParameter	A required parameter for the specified action is not supplied.	400	忘记输入参数，具体参数名在错误消息中说明 (UserName)
InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter.	400	参数格式/取值范围不对，具体参数名在错误消息中说明 (UserName, Path, RealName)
EntityAlreadyExists	The request was rejected because it attempted to create a resource that already exists	409	资源已经存在，具体实体名称在错误消息中说明 (UserName)
LimitExceeded	The request was rejected because it attempted to create resources beyond the current Ksyun account limits	409	资源数量超过限制，具体限制在错误消息中说明 (User数量是100)

示例（调用成功）

请求示例

https://iam.api.ksyun.com/?Action=CreateVirtualMFADevice&Version=2015-11-01&VirtualMFADeviceName=huangxin&公共请求参数

返回示例

- XML格式

```

<CreateVirtualMFADeviceResponse>
  <CreateVirtualMFADeviceResult>
    <VirtualMFADevice>
      <SerialNumber>krn:ksc:iam::2000003616:mfa/huangxin</SerialNumber>
      <Base32StringSeed>QTZYnk5aSzdMSUhENEZIUzNGQ1E0W1RTQ01NSFNSTUJDUNPNQEQ3NEhDQ1FZQVFTENEQko0NEdHU1pPR1hWQg==</Base32StringSeed>
      <QrcodePNG>...</QrcodePNG>
    </VirtualMFADevice>
  </CreateVirtualMFADeviceResult>
  <ResponseMetadata>
    <RequestId>b9dala9e-9a7f-4c55-a92a-5bca05caa209</RequestId>
  </ResponseMetadata>
</CreateVirtualMFADeviceResponse>

```

- JSON示例

```
{
  "CreateVirtualMFADeviceResult": {
    "VirtualMFADevice": {
      "SerialNumber": "krn:ksc:iam::2000003616:mfa/huangxin1",
      "Base32StringSeed": "UjzXTUo3Q0M1VEp0WFdCNFpBNFdMNURUUFBNV1hCSk1IIEVaRVdNUVhXNk00M1ZLRTVON0xDS0JHSTIyNFdBTw==",
      "QrcodePNG": "...",
    }
  },
  "RequestId": "51389638-7d76-486c-9428-d0263cccd227"
}
```

查询虚拟设备列表

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是	确定值	[a-zA-Z]+	操作接口名，系统规定参数，取值：ListVirtualMFADevices
AssignmentStatus	String	是	确定值	-	Assigned表示已分配, Unassigned表示未分配, Any表示所有。默认值为Any
Marker	String	否	最短1，最长320		当返回结果被截断时，使用本次返回的Marker值用于标示下次调用查询操作的起始点，Marker值不会过期
MaxItems	Integer	否	最小值1，最大值1000，默认100		用于限制本次查询结果返回的数量，如果仍有额外元素未显示，则返回元素中的IsTruncated的值会被设置为true，同时返回下次查询的起始点Marker值

错误码（调用不成功）

错误代码	描述	HTTP 状态码	语义
MissingParameter	A required parameter for the specified action is not supplied.	400	忘记输入参数，具体参数名在错误消息中说明 (UserName)
InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter.	400	参数格式/取值范围不对，具体参数名在错误消息中说明 (UserName, Path, RealName)
EntityAlreadyExists	The request was rejected because it attempted to create a resource that already exists	409	资源已经存在，具体实体名称在错误消息中说明 (UserName)
LimitExceeded	The request was rejected because it attempted to create resources beyond the current Ksyun account limits	409	资源数量超过限制，具体限制在错误消息中说明 (User数量是100)

示例（调用成功）

请求示例

https://iam.api.ksyun.com/?Action=ListVirtualMFADevices&Version=2015-11-01&公共请求参数

返回示例

- XML格式

```
<ListVirtualMFADevicesResponse>
  <ListVirtualMFADevicesResult>
    <IsTruncated></IsTruncated>
    <VirtualMFADevices>
      <member>
        <Item>
          <SerialNumber>krn:ksc:iam::2000003616:mfa/huangxin1</SerialNumber>
        </Item>
        <Item>
          <SerialNumber>krn:ksc:iam::2000003616:mfa/huangxin</SerialNumber>
        </Item>
        <Item>
          <SerialNumber>krn:ksc:iam::2000003616:mfa/test</SerialNumber>
        </Item>
      </member>
    </VirtualMFADevices>
  </ListVirtualMFADevicesResult>
```

```

<ResponseMetadata>
  <RequestId>07fc6c4a-9842-4c57-b080-9d53577e6484</RequestId>
</ResponseMetadata>
</ListVirtualMFADevicesResponse>

```

- JSON示例

```

{
  "ListVirtualMFADevicesResult": {
    "IsTruncated": false,
    "VirtualMFADevices": {
      "member": [
        {
          "SerialNumber": "krn:ksc:iam::2000003616:mfa/huangxin1"
        },
        {
          "SerialNumber": "krn:ksc:iam::2000003616:mfa/huangxin"
        },
        {
          "SerialNumber": "krn:ksc:iam::2000003616:mfa/test"
        }
      ]
    }
  },
  "RequestId": "e92d55c3-79ec-44a2-b932-8ac5f2f321f4"
}

```

激活虚拟设备

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是	确定值	[a-zA-Z]+	操作接口名，系统规定参数，取值：EnableMFADevice
AuthenticationCode1	Integer	是	6位		MFA六位码1
AuthenticationCode2	Integer	是	6位		MFA六位码2
SerialNumber	String	是	最短1, 最长1000		被绑定的MFA设备序列号，如果是虚拟MFA设备，则是该虚拟MFA设备的krn
UserName	String	是	最短1, 最长64	^[\\w\\Q+,.@-\\E]+	被绑定的子账号用户名

错误码（调用不成功） {#错误码（调用不成功）}

错误代码	描述	HTTP 状态码	语义
MissingParameter	A required parameter for the specified action is not supplied.	400	忘记输入参数，具体参数名在错误消息中说明(UserName)
InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter.	400	参数格式/取值范围不对，具体参数名在错误消息中说明(UserName, Path, RealName)
EntityAlreadyExists	The request was rejected because it attempted to create a resource that already exists	409	资源已经存在，具体实体名称在错误消息中说明(UserName)
LimitExceeded	The request was rejected because it attempted to create resources beyond the current Ksyun account limits	409	资源数量超过限制，具体限制在错误消息中说明(User数量是100)

示例（调用成功）

请求示例

https://iam.api.ksyun.com/?Action=EnableMFADevice&Version=2015-11-01&公共请求参数

返回示例

- JSON示例

```

{
  "ResponseMetadata": {

```

```

    "RequestId": "9b89c62d-64fa-470c-8a41-69326c0e95a7"
  }
}

```

解绑虚拟设备

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是	确定值	[a-zA-Z]+	操作接口名，系统规定参数，取值：DeactivateMFADevice
SerialNumber	String	是	最短1, 最长1000		被绑定的MFA设备序列号，如果是虚拟MFA设备，则是该虚拟MFA设备的krn
UserName	String	是	最短1, 最长64	^[\\w\\Q+=,\\.@-\\E]+	被绑定的子账号用户名

错误码（调用不成功） {#错误码（调用不成功）}

错误代码	描述	HTTP 状态码	语义
MissingParameter	A required parameter for the specified action is not supplied.	400	忘记输入参数，具体参数名在错误消息中说明(UserName)
InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter.	400	参数格式/取值范围不对，具体参数名在错误消息中说明(UserName, Path, RealName)
EntityAlreadyExists	The request was rejected because it attempted to create a resource that already exists	409	资源已经存在, 具体实体名称在错误消息中说明(UserName)
LimitExceeded	The request was rejected because it attempted to create resources beyond the current Ksyun account limits	409	资源数量超过限制，具体限制在错误消息中说明(User数量是100)

示例（调用成功）

请求示例

https://iam.api.ksyun.com/?Action=DeactivateMFADevice&Version=2015-11-01&公共请求参数

返回示例

- JSON示例

```

{
  "ResponseMetadata": {
    "RequestId": "9b89c62d-64fa-470c-8a41-69326c0e95a7"
  }
}

```

删除虚拟设备

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是	确定值	[a-zA-Z]+	操作接口名，系统规定参数，取值：DeactivateMFADevice
SerialNumber	String	是	最短1, 最长1000		被绑定的MFA设备序列号，如果是虚拟MFA设备，则是该虚拟MFA设备的krn
UserName	String	是	最短1, 最长64	^[\\w\\Q+=,\\.@-\\E]+	被绑定的子账号用户名

错误码（调用不成功） {#错误码（调用不成功）}

错误代码	描述	HTTP 状态码	语义
------	----	----------	----

MissingParameter	A required parameter for the specified action is not supplied.	400	忘记输入参数，具体参数名在错误消息中说明(UserName)
InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter.	400	参数格式/取值范围不对，具体参数名在错误消息中说明(UserName, Path, RealName)
EntityAlreadyExists	The request was rejected because it attempted to create a resource that already exists	409	资源已经存在，具体实体名称在错误消息中说明(UserName)
LimitExceeded	The request was rejected because it attempted to create resources beyond the current Ksyun account limits	409	资源数量超过限制，具体限制在错误消息中说明(User数量是100)

示例（调用成功）

请求示例

https://iam.api.ksyun.com/?Action=DeactivateMFADevice&Version=2015-11-01&公共请求参数

返回示例

- JSON示例

```
{
  "ResponseMetadata": {
    "RequestId": "9b89c62d-64fa-470c-8a41-69326c0e95a7"
  }
}
```

获取虚拟设备

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是	确定值	[a-zA-Z]+	操作接口名，系统规定参数，取值：GetVirtualMFADevice
UserName	String	是	最短1, 最长64	^[\\w\\Q+,.\\@-\\E]+	用户名

错误码（调用不成功）

错误代码	描述	HTTP 状态码	语义
MissingParameter	A required parameter for the specified action is not supplied.	400	忘记输入参数，具体参数名在错误消息中说明(UserName)
InvalidParameterValue	An invalid or out-of-range value was supplied for the input parameter.	400	参数格式/取值范围不对，具体参数名在错误消息中说明(UserName, Path, RealName)
EntityAlreadyExists	The request was rejected because it attempted to create a resource that already exists	409	资源已经存在，具体实体名称在错误消息中说明(UserName)
LimitExceeded	The request was rejected because it attempted to create resources beyond the current Ksyun account limits	409	资源数量超过限制，具体限制在错误消息中说明(User数量是100)

示例（调用成功）

请求示例

https://iam.api.ksyun.com/?Action=GetVirtualMFADevice&Version=2015-11-01&公共请求参数

返回示例

- XML格式

```
<GetVirtualMFADeviceResponse>
  <GetVirtualMFADeviceResult>
    <VirtualMFADevice>
```

```

<Item>
  <SerialNumber>krn:ksc:iam::2000000719:mfa/mfa</SerialNumber>
  <EnableDate>2017-11-03T10:58:51+0800</EnableDate>
  <User>
    <UserId>t160DMvLRsCqe9EgrM-B2g</UserId>
    <UserName>mfa</UserName>
    <Krn>krn:ksc:iam::2000000719:usermfa</Krn>
    <CreateDate>2017-11-03T10:57:53+0800</CreateDate>
  </User>
</Item>

</VirtualMFADevice>
<ResponseMetadata>
  <RequestId>2</RequestId>
</ResponseMetadata>
</EnableMFADeviceResponse>

```

创建项目

请求参数

名称	类型	必填	长度限制（字符）	参数格式	描述
Action	String	是	确定值	[a-zA-Z]+	操作接口名，系统规定参数，取值：CreateProject
Version	String	是	确定值		固定为：2015-11-01
ProjectName	String	是	64个字符以内		项目名称
ProjectDesc	String	否	64个字符以内		项目描述

请求示例

```

curl -X POST 'http://iam.api.ksyun.com'
-H 'host: iam.api.ksyun.com'
-H 'authorization: AWS4-HMAC-SHA256 Credential=ak/20190426/region/iam/aws4_request, SignedHeaders=host;x-amz-date, Signature='
-H 'content-type: application/x-www-form-urlencoded'
-H 'x-amz-date: 20190426T091629Z'
-d Action=CreateProject&Version=2015-11-01&ProjectName=x&ProjectDesc=x

```

返回结果集样例

```

{
  "Result":xxx, //项目ID
  "RequestId":"2c95091b-6251-4a08-ad0e-1d378cd5587e"
}

```

获取主账号/子用户下所有项目列表

获取主账号/子用户下所有项目列表(针对子用户的访问，仅返回该子用户是项目成员的项目列表) 返回的列表按照创建时间降序排序

请求参数

名称	类型	必填	长度限制（字符）	参数格式	描述
Action	String	是	确定值	[a-zA-Z]+	操作接口名，系统规定参数，取值：GetAccountAllProjectList
Version	String	是	确定值		固定为：2015-11-01

请求示例

<http://iam.api.ksyun.com/?Action=GetAccountAllProjectList>&公共请求参数

返回结果集样例

```

{
  "ListProjectResult": {
    "Total": 2,
    "ProjectList":
    [
      {
        "ProjectId": 100001,
        "AccountId": "73401119",

```

```

    "ProjectName": "test21",
    "ProjectDesc": "123456789012345678901234567890",
    "CreateTime": "2017-08-08 20:48:46"
  },
  {
    "ProjectId": 0,
    "AccountId": "73401119",
    "ProjectName": "默认项目",
    "ProjectDesc": "默认项目",
    "CreateTime": "2017-08-08 20:48:46"
  }
]
},
"RequestId": "31b7414b-ab1f-45dd-aa87-9549989fcc18"
}

```

查询项目成员

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是	确定值	[a-zA-Z]+	操作接口名，系统规定参数，取值：ListProjectMember
Version	String	是	确定值	YYYY-MM-DD	接口版本号，版本号不同接口支持的参数和返回值可能不同，身份与访问管理接口当前只支持一个版本，即2015-11-01
ProjectId	int	是			项目id

示例

请求示例

http://iam.api.ksyun.com/?Version=2015-11-01&Action=ListProjectMember&ProjectId=01&公共请求参数

返回示例

其中IdentityType：1代表子用户，2代表角色。

- XML格式

```

<ListProjectMemberResponse>
  <ListProjectMember>
    <Item>
      <MemberId>1111</MemberId>
      <IdentityType>2</IdentityType>
      <IdentityName>role1</IdentityName>
      <CreateTime>2021-03-23 11:51:42</CreateTime>
      <Krn>krn:ksc:iam::ksc:role/role1</Krn>
      <ServiceRoleType></ServiceRoleType>
    </Item>
    <Item>
      <MemberId>2222</MemberId>
      <IdentityType>1</IdentityType>
      <IdentityName>test1</IdentityName>
      <CreateTime>2020-07-28 19:00:20</CreateTime>
      <Krn>krn:ksc:iam::ksc:user/test1</Krn>
    </Item>
  </ListProjectMember>
  <ResponseMetadata>
    <RequestId>f8f4c72b-aa5b-4a48-85ac-5217feb1d70c</RequestId>
  </ResponseMetadata>
</ListProjectMemberResponse>

```

- JSON示例

```

{
  "ListProjectMember": [
    {
      "MemberId": "1111111",
      "IdentityType": "2",
      "IdentityName": "role1",
      "CreateTime": "2021-03-23 11:51:42",
      "Krn": "krn:ksc:iam::ksc:role/role1",
      "ServiceRoleType": null
    }
  ]
}

```



```

    },
    {
      "MemberId": 222222,
      "IdentityType": 1,
      "IdentityName": "test1",
      "CreateTime": "2020-07-28 19:00:20",
      "Krn": "krn:ksc:iam::ksc:user/test1"
    }
  ],
  "RequestId": "f4332011-3a97-48f2-8ea7-c819d998bfba"
}

```

移除项目成员

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是	确定值	[a-zA-Z]+	操作接口名，系统规定参数，取值：DeleteProjectMember
Version	String	是	确定值	YYYY-MM-DD	接口版本号，版本号不同接口支持的参数和返回值可能不同，身份与访问管理接口当前只支持一个版本，即2015-11-01
ProjectId	int	是			项目id
MemberIds	int	是			要移除的成员id

错误码（调用不成功）

错误代码(Code)	错误消息(Message)	HTTP 状态码	中文描述（语义）
NotExistProjectId	project id not exist	400	项目id不存在
ProjectMemberNotExist	The request was rejected because the user is not this project member	400	请求被拒绝，被移除的成员不属于这个项目

示例

请求示例

http://iam.api.ksyun.com/?Version=2015-11-01&Action=DeleteProjectMember&ProjectId=0&MemberIds=1111&公共请求参数

返回示例

- XML格式

```

<DeleteProjectMemberResponse>
  <Result>1</Result>
  <ResponseMetadata>
    <RequestId>03c6576b-4fc4-4b2a-af29-8b3176fbb25e</RequestId>
  </ResponseMetadata>
</DeleteProjectMemberResponse>

```

- JSON示例

```

{
  "Result": true,
  "RequestId": "61fa4124-0cbc-495f-b859-140bc7121dfc"
}

```

获取项目里面资源列表

根据指定的项目id，获取相应的资源列表

HTTP METHOD : GET

请求参数

名称	类型	必填	长度限制 (字符)	参数格式	描述
Action	String	是	确定值	[a-zA-Z]+	操作接口名, 系统规定参数, 取值: GetProjectInstanceList
Version	String	是	确定值		固定为: 2015-11-01
ProjectId	String	否			项目id, 默认为0
Pn	Int	否			页码。默认1
Ps	Int	否			每页条数, 默认10, 最大100条

请求示例

<https://iam.api.ksvun.com/?Action=GetProjectInstanceList&ProjectId=0&Pn=1&>公共请求参数

返回结果集样例

```
{
  "ListInstanceResult": {
    "Items": [
      {
        "ProjectId": 0,
        "InstanceId": "8baaf472-ef0a-4da4-b97e-22d7a4562e84",
        "ProductGroup": 101,
        "ProductType": 4,
        "Region": "cn-shanghai-3",
        "AccountId": 73401119,
        "Status": 3,
        "ServiceBeginTime": "2016-08-09 15:39:54",
        "ServiceEndTime": "2016-11-30 23:59:59"
      }
    ],
    "CurrentPage": 1,
    "PageSize": 1,
    "Total": 7079
  },
  "RequestId": "4e750956-2a93-42c2-a484-bcdef5b3ccdb"
}
```

错误码

错误码	响应码	错误信息
ProjectMemberNotExist	400	子用户不是该项目的成员
NotExistProjectId	400	项目id非法

批量查询资源所属的项目ID列表

根据指定的资源ID列表获取对应的项目ID列表

HTTP METHOD : POST

请求参数

名称	类型	必填	长度限制 (字符)	参数格式	描述
Action	String	是	确定值	[a-zA-Z]+	操作接口名, 系统规定参数, 取值: BatchGetInstanceProjectInfo
Version	String	是	确定值		固定为: 2015-11-01
InstanceIds	String	是			实例id, 可以支持多个, 以半角逗号分隔, 最多500个

请求示例

<https://iam.api.ksvun.com/?Action=BatchGetInstanceProjectInfo&&InstanceIds=xxx&>公共请求参数

返回结果集样例

```
{
  "ListProjectResult": [
    {
      "InstanceId": "InstanceId1",
    }
  ]
}
```

```

    "ProjectId": 0
  },
  {
    "InstanceId": "InstanceId2",
    "ProjectId": 10023
  }
],
"RequestId": "31b7414b-ab1f-45dd-aa87-9549989fcc18"
}

```

错误码

错误码	响应码	错误信息
InvalidParameter	400	请求字段不合法，会包含具体的非法字段，和具体的错误信息

查询账户概要信息

请求参数

名称	类型	必须	长度限制(字符)	参数格式	描述
Action	String	是	确定值	[a-zA-Z]+	操作接口名，系统规定参数，取值：GetAccountSummary

返回元素（调用成功）

名称	类型	描述
SummaryMap	SummaryMap Type	IAM概要信息映射表

错误码（调用不成功）

无

示例（调用成功）

请求示例

https://iam.api.ksyun.com/?Action=GetAccountSummary
&公共请求参数

返回示例

- XML格式

```

<GetAccountSummaryResponse>
  <ResponseMetadata>
    <RequestId>b1e84433-3110-4357-b3bc-1ebb2d87656a</RequestId>
  </ResponseMetadata>
  <GetAccountSummaryResult>
    <SummaryMap>
      <entry>
        <key>AccessKeysPerUserQuota</key>
        <value>2</value>
      </entry>
      <entry>
        <key>AccountAccessKeysPresent</key>
        <value>1</value>
      </entry>
      <entry>
        <key>AttachedPoliciesPerUserQuota</key>
        <value>5</value>
      </entry>
      <entry>
        <key>Policies</key>
        <value>0</value>
      </entry>
      <entry>
        <key>PoliciesQuota</key>
        <value>50</value>
      </entry>
      <entry>
        <key>PolicySizeQuota</key>
        <value>2048</value>
      </entry>
    </SummaryMap>
  </GetAccountSummaryResult>
</ResponseMetadata>
</GetAccountSummaryResponse>

```

```

    <entry>
      <key>PolicyVersionsInUse</key>
      <value>2</value>
    </entry>
  <entry>
    <key>PolicyVersionsInUseQuota</key>
    <value>5</value>
  </entry>
  <entry>
    <key>Users</key>
    <value>5</value>
  </entry>
  <entry>
    <key>UsersQuota</key>
    <value>100</value>
  </entry>
  <entry>
    <key>VersionsPerPolicyQuota</key>
    <value>5</value>
  </entry>
</SummaryMap>
</GetAccountSummaryResult>
</GetAccountSummaryResponse>

```

- JSON示例

```

{
  "ResponseMetadata": {
    "RequestId": "bb369d0d-4183-45c4-8177-2e7e09f37d7b"
  },
  "GetAccountSummaryResult": {
    "SummaryMap": {
      "entry": [
        {
          "key": "AccessKeysPerUserQuota",
          "value": "2"
        },
        {
          "key": "AccountAccessKeysPresent",
          "value": "1"
        },
        {
          "key": "AttachedPoliciesPerUserQuota",
          "value": "5"
        },
        {
          "key": "Policies",
          "value": "0"
        },
        {
          "key": "PoliciesQuota",
          "value": "50"
        },
        {
          "key": "PolicySizeQuota",
          "value": "2048"
        },
        {
          "key": "PolicyVersionsInUse",
          "value": "2"
        },
        {
          "key": "PolicyVersionsInUseQuota",
          "value": "5"
        },
        {
          "key": "Users",
          "value": "5"
        },
        {
          "key": "UsersQuota",
          "value": "100"
        },
        {
          "key": "VersionsPerPolicyQuota",
          "value": "5"
        }
      ]
    }
  }
}

```

User

描述

用户数据类型，包含IAM用户实体的相关信息。在如下操作中作为响应元素被使用：

- [CreateUser](#)
- [GetUser](#)
- [ListUsers](#)

内容

名称	类型	必须	长度限制(字符)	参数格式	描述
Krn	String	是	最短是20, 最长2048	格式: <code>krn:ksc:iam::account-id:user/user-name</code>	用户的唯一标识, 使用KRN
CreateDate	DateTime	是	20	按照ISO8601标准, 使用UTC时间, 格式为"YYYY-MM-DDThh:mm:ssZ"例如, 2014-05-26T12:00:00.000Z (为北京时间2014年5月26日20点0分0秒0毫秒)	用户的创建时间
Path	String	是	最短1, 最长512	/	路径
UserId	String	是	22字符	<code>[a-zA-Z0-9-_]+</code>	用户唯一标识
UserName	String	是	最短1, 最长64	<code>[\w+=[. @-]+</code>	新创建的IAM用户的用户名
RealName	String	是	最短2, 最长128	<code>[\u4e00-\u9fff]+</code>	IAM用户的真实姓名

Role

描述

角色数据类型，包含角色的相关信息。在如下操作中作为响应元素被使用：

- [CreateRole](#)
- [GetRole](#)
- [ListRoles](#)

内容

名称	类型	必须	长度限制(字符)	参数格式	描述
Krn	String	是	最短是25, 最长2048	格式: <code>krn:ksc:iam::account-id:role/role-name</code>	角色的唯一标识
CreateDate	DateTime	是	20	按照ISO8601标准, 使用UTC时间, 格式为"YYYY-MM-DDThh:mm:ssZ"例如, 2014-05-26T12:00:00Z (为北京时间2014年5月26日20点0分0秒)	创建时间
Path	String	是	最短1, 最长512	<code>\</code> or <code>\[:graph:]+\</code>	路径
RoleId	String	是	22字符	<code>[a-zA-Z0-9-_]+</code>	Role唯一标识
RoleName	String	是	最短1, 最长64	<code>[\w+=[. @-]+</code>	角色名
TrustedAccounts	String	是	最短1, 最长2048	<code>[\w\s:~*{} []/\$-]+</code>	描述角色的信任关系的策略文档

LoginProfile

描述

登录配置数据类型，包含IAM用户实体的登录配置信息，包括用户名、密码创建时间和是否下次登录设置新密码。在如下操作中作为响应元素被使用：

- [CreateLoginProfile](#)
- [GetLoginProfile](#)

注：用户的登录密码属于登录配置的组成元素，需要被安全可靠保存，但不会任何方式显示给用户。

内容

名称	类型	必须	长度限制(字符)	参数格式	描述
Createdate	DateTime	是	20	按照ISO8601标准，使用UTC时间，格式为“YYYY-MM-DDThh:mm:ssZ”例如，2014-05-26T12:00:00.000Z（为北京时间2014年5月26日20点0分0秒0毫秒）	登录配置信息的创建时间
PasswordResetRequired	Boolean	是		true/false	用户下次成功登录后是否需要设置新密码，默认false，即不设定
UserName	String	是	最短1，最长64	[\w+=,.\@-]+	登录配置所属的IAM用户的用户名

AccessKey

描述

访问密钥数据类型，包含访问密钥的相关信息。在如下操作中作为响应元素被使用：

- [CreateAccessKey](#)

内容

名称	类型	必须	长度限制(字符)	参数格式	描述
Createdate	DateTime	是	20	按照ISO8601标准，使用UTC时间，格式为“YYYY-MM-DDThh:mm:ssZ”例如，2014-05-26T12:00:00.000Z（为北京时间2014年5月26日20点0分0秒0毫秒）	访问密钥的创建时间
AccessKeyId	String	是	最短20，最长32	AKLT[a-zA-Z0-9-]+	访问密钥ID
SecretAccessKey	String	是	68	[a-zA-Z0-9/+]==	私有访问密钥
Status	String	是	6或8	Active/Inactive	访问密钥的状态，Active为可用，Inactive为不可用
UserName	String	否	最短1，最长64	[\w+=,.\@-]+	如果是IAM用户的访问密钥，该域显示其用户名；如果是主账户的访问密钥，且已经设置别名，则该域显示别名信息，否则不显示

AccessKeyMetadata

描述

访问密钥元数据数据类型，不包含私有访问密钥的访问密钥相关信息。在如下操作中作为响应元素被使用：

- [ListAccessKeys](#)

内容

名称	类型	必须	长度限制(字符)	参数格式	描述
----	----	----	----------	------	----

CreateDate	DateTime	是	20	按照ISO8601标准，使用UTC时间，格式为“YYYY-MM-DDThh:mm:ssZ”例如，2014-05-26T12:00:00.000Z（为北京时间2014年5月26日20点0分0秒0毫秒）	访问密钥的创建时间
AccessKeyId	String	是	最短20, 最长32	AKLT[a-zA-Z0-9_]+	访问密钥ID
Status	String	是	6或8	Active/Inactive	访问密钥的状态，Active为可用，Inactive为不可用
UserName	String	否	最短1, 最长64	[\w+=, .@-]+	如果是IAM用户的访问密钥，该域显示其用户名；如果是主账户的访问密钥，且已经设置别名，则该域显示别名信息，否则不显示

Policy

描述

策略数据类型，包含策略实体的元数据信息。在如下操作中作为响应元素被使用：

- [CreatePolicy](#)
- [GetPolicy](#)
- [ListPolicies](#)

内容

名称	类型	必须	长度限制(字符)	参数格式	描述
Krn	String	是	最短是25, 最长2048	格式: krn:ksc:iam:: <i>account-id</i> :policy/ <i>policy-name</i>	策略的唯一标识，使用KRN
CreateDate	DateTime	是	20	按照ISO8601标准，使用UTC时间，格式为“YYYY-MM-DDThh:mm:ssZ”例如，2014-05-26T12:00:00.000Z（为北京时间2014年5月26日20点0分0秒0毫秒）	策略的创建时间，即第一个策略版本的创建时间
UpdateDate	DateTime	是	20	按照ISO8601标准，使用UTC时间，格式为“YYYY-MM-DDThh:mm:ssZ”例如，2014-05-26T12:00:00.000Z（为北京时间2014年5月26日20点0分0秒0毫秒）	策略上次更新时间，如果只有一个策略版本，更新时间与创建时间一致，否则更新时间是最后一个策略版本的创建时间
Path	String	是	最短1, 最长512	/	策略的路径
PolicyId	String	是	22字符	[a-zA-Z0-9_]+	策略唯一标识
PolicyName	String	是	最短1, 最长128	[\w+=, .@-]+	策略名称
DefaultVersionId	String	是	最短2	v[1-9][0-9]*	策略的默认版本Id
AttachmentCount	Integer	是		数值	附加该策略的实体数量，目前只有一种实体IAM用户
Description	String	否	0-1000		策略的友好描述信息，GetPolicy操作返回该元素，CreatePolicy和ListPolicies操作不返回该元素

PolicyVersion

描述

策略版本数据类型，包含策略版本的信息。在如下操作中作为响应元素被使用：

- [CreatePolicyVersion](#)
- [GetPolicyVersion](#)
- [ListPolicyVersions](#)

内容

名称	类型	必须	长度限制(字符)	参数格式	描述
CreateDate	DateTime	是	20	按照ISO8601标准, 使用UTC时间, 格式为"YYYY-MM-DDThh:mm:ssZ"例如, 2014-05-26T12:00:00.000Z (为北京时间2014年5月26日20点0分0秒0毫秒)	策略版本的创建时间
Document	String	否	最短1, 最长5K	<code>[\w\s:"*{}[]/\$-]+</code>	策略版本的策略文档内容, 只有调用GetPolicyVersion接口时显示该元素
VersionId	String	是	最短2	<code>v[1-9][0-9]*</code>	策略版本Id
IsDefaultVersion	Boolean	是			用于指定当前策略版本是否为默认策略版本

AttachedPolicy

描述

附加策略数据类型, 包含附加策略实体的元数据信息。在如下操作中作为响应元素被使用:

- [ListAttachedUserPolicies](#)

内容

名称	类型	必须	长度限制(字符)	参数格式	描述
PolicyKrn	String	是	最短是25, 最长2048	格式: <code>krn:ksc:iam::account-id:policy/policy-name</code>	策略的唯一标识, 使用KRN
PolicyName	String	是	最短1, 最长128	<code>[\w+=,.\@-]+</code>	策略名称

PolicyUser

描述

策略关联用户数据类型, 包含指定策略的关联用户实体的元数据信息。在如下操作中作为响应元素被使用:

- [ListEntitiesForPolicy](#)

内容

名称	类型	必须	长度限制(字符)	参数格式	描述
UserName	String	是	最短1, 最长64	<code>[\w+=,.\@-]+</code>	策略关联用户名

SummaryMap

描述

概要映射表数据类型, 包含IAM实体的使用和配额信息。在如下操作中作为响应元素被使用:

- [GetAccountSummary](#)

内容

名称	类型	必须	长度限制(字符)	参数格式	描述
AccessKeysPerUserQuota	Integer	是	无	数字	每个IAM用户的访问密钥配额, 2
AccountAccessKeysPresent	Integer	是	无	0或1	账户当前是否有访问密钥, 有是1, 否则是0
AttachedPoliciesPerUserQuota	Integer	是	无	数字	每个IAM用户可附加的策略数量配额, 5

Policies	Integer	是	无	数字	账户当前自定义策略的数量
PoliciesQuota	Integer	是	无	数字	账户的自定义策略数量配额, 50
PolicySizeQuota	Integer	是	无	数字	自定义策略的字符数配额, 不包括空白字符, 2048
PolicyVersionsInUse	Integer	是	无	数字	账户当前附加到IAM实体的策略数量
PolicyVersionsInUseQuota	Integer	是	无	数字	账户中可附加到IAM实体的策略数量配额, 500
Users	Integer	是	无	数字	账户当前IAM用户数量
UsersQuota	Integer	是	无	数字	账户的IAM用户数量配额, 100
VersionsPerPolicyQuota	Integer	是	无	数字	自定义策略的版本数量配额, 5

PasswordPolicy

描述

密码策略数据类型, 包含IAM用户密码策略的详细信息。在如下操作中作为响应元素被使用:

- [GetAccountPasswordPolicy](#)

内容

名称	类型	必须	长度限制(字符)	参数格式	描述
AllowUsersToChangePassword	Boolean	是	无	无	是否允许全部IAM用户在控制台修改自己的密码
ExpirePasswords	Boolean	是	无	无	是否允许密码过期, 如果MaxPasswordAge大于0则取值true, 否则false
HardExpiry	Boolean	是	无	无	是否阻止IAM用户在密码过期后修改自己的密码
MaxPasswordAge	Integer	否	最小值1, 最大值1095	无	IAM用户密码有效天数, 默认0则不显示
MinimumPasswordLength	Integer	是	最小值8, 最大值128	无	IAM用户密码最小长度
PasswordReusePrevention	Integer	否	最小值1, 最大值24	无	阻止IAM用户使用的历史密码数量, 默认0则不显示
RequireLowercaseCharacters	Boolean	是	无	无	是否强制要求IAM用户密码包含一个小写字母
RequireNumbers	Boolean	否	无	无	是否强制要求IAM用户密码包含一个数字
RequireSymbols	Boolean	否	无	无	是否强制要求IAM用户密码包含一个后面的符号 (! @ # \$ % ^ & * () _ + - = [] { } ')
RequireUppercaseCharacters	Boolean	否	无	无	是否强制要求IAM用户密码包含一个大写字母

VirtualMFADevice

描述

虚拟多因素认证数据类型, 包含虚拟多因素认证设备的相关信息。

在如下操作中作为响应元素被使用:

- [CreateVirtualMFADevice](#)

内容 {

名称	类型	必须	长度限制(字符)	参数格式	描述
SerialNumber	String	是	20	[\w+=/:\. @-]+	设备序列号

Base32StringSeed	String	否	最短20, 最长32	Base64-encoded binary data	多因素认证设备密钥
QRCodePNG	String	否	68	Base64-encoded binary data	密钥二维码PNG
EnabledDate	DateTime	否	20	按照ISO8601标准, 使用UTC时间, 格式为"YYYY-MM-DDThh:mm:ssZ"例如, 2014-05-26T12:00:00Z (为北京时间2014年5月26日20点0分0秒)	虚拟多因素认证设备激活日期
User	User数据类型	否	最短1, 最长64	[\w+=[. @-]+	绑定用户的基本信息

AssumeRoleResult

AssumeRoleResult {#73-assumeroleresult}

描述

访问密钥数据类型, 包含临时访问密钥的相关信息。在如下操作中作为响应元素被使用:

- [AssumeRole](#)

AssumedRoleUser内容

名称	类型	必须	长度限制(字符)	参数格式	描述
Arn	String	是	min:20, max:2048	krn:ksc:sts::account-id:assumed-role/role-name/role-session-name	
AssumedRoleId	String	是	min:2, max:96	[\w+=[. @-:]*+	包含roleId 和 role_session_name, 用冒号: 做分隔

Credentials内容

名称	类型	必须	长度限制(字符)	参数格式	描述
AccessKeyId	String	是	min:20, max:32	AKRT[a-zA-Z0-9-_*]	
Expiration	Timestamp	是			当前临时证书的过期时间
SecretAccessKey	String	是		[a-zA-Z0-9/+]=*	
SecurityToken	String	是			会话session

其他内容

名称	类型	必须	长度限制(字符)	参数格式	描述
PackedPolicySize	String	是	min:0		返回0

- JSON示例

```
{
  "AssumeRoleResult": {
    "Credentials": {
      "SecretAccessKey": "xxxx==",
      "Expiration": "2018-06-19T10:32:50Z",
      "AccessKeyId": "AKTRxxxx",
      "SecurityToken": "V1KYiiBxLgb0U2tI3hBLVNL7jq1nfj+6NDubugqHTjSKzUxMPQoQF9LWBCN/JoTVaJL4AoL2xB7v7MR0ibed+7+KQ95IrocgRpF0wotCW+UiBbcgCGlUZlvgJgY4aXj/u9SNCu0TITgZFFFmCJBND2iagzkiCiKzIvFHuPesYrcBMN3Na2IdtTIEtM04GPr4d3PzMYE6cuSs900EHCsiQ05e8kfyYNGGHUj7RykEzA711HuNv7m8U00JncNGAmbeXMaXp0enyTWkzJJYN+FLT5BoXFJ3RWCYI19byd8Ub8Z7T5TXsggIJ+C+GfhuatH7Gk5a68bCnoMETUzMQDDA1lg=="
    },
    "AssumedRoleUser": {
      "Krn": "krn:ksc:sts::xxx:assumed-role/xxx/thisisatest",
      "AssumedRoleId": "xxx:thisisatest"
    }
  },
  "RequestId": "93e14aef-442f-455f-bbcc-fecec40fe364"
}
```

UserSession

描述

用户数据类型，包含IAM用户实体的相关信息。在如下操作中作为响应元素被使用：

- [GetUserSession](#)

内容

名称	类型	必须	长度限制(字符)	参数格式	描述
Url	String	是	最短是20, 最长2048	-	获取Session校验的地址
Expiration	String	是	20个字符	Y-m-d\TH:i:s\Z	连接的失效时间

KRN查询列表

资源名称	英文名称	资源KRN
实例	instance	karn:ksc:kec:region:account-id:instance/instance-id
云主机镜像	image	karn:ksc:kec:region::image/image-id
卷	volume	karn:ksc:kec:region:account-id:volume/volume-id
快照	snapshot	karn:ksc:kec:region:account-id:snapshot/snapshot-id
安全组	security-group	karn:ksc:vpc:region:account-id:security-group/security-group-id
子网	subnet	karn:ksc:vpc:region:account-id:subnet/subnet-id
网络接口	network-interface	karn:ksc:vpc:region:account-id:network-interface/network-interface-id
虚拟私有网络	vpc	karn:ksc:vpc:region:account-id:vpc/vpc-id
网络ACL	network-acl	karn:ksc:vpc:region:account-id:network-acl/network-acl-id
路由	route	karn:ksc:vpc:region:account-id:route/route-id
本地地址转换	nat	karn:ksc:vpc:region:account-id:nat/nat-id
隧道网关	tunnel	karn:ksc:vpc:region:account-id:tunnel/tunnel-id
对等连接	vpc-peering-connection	karn:ksc:vpc:region:account-id:vpc-peering-connection/vpc-peering-connection-id
负载均衡器	loadbalancer	karn:ksc:slb:region:account-id:loadbalancer/load-balancer-id
监听器	listener	karn:ksc:slb:region:account-id:listener/listener-id
主账户	root	karn:ksc:iam::account-id:root
用户	user	karn:ksc:iam::account-id:user/user-name
角色	role	karn:ksc:iam::account-id:role/role-name
assumeRole	assumeRole	karn:ksc:sts::account-id:assumed-role/role-name/role-session-name
策略	policy	karn:ksc:iam::account-id ksc:policy/policy-name
云物理主机镜像	image	karn:ksc:epc:region::image/image-id
云物理主机实例	epc-host	karn:ksc:epc:region:account-id:epc-host/epc-host-id
域名	domain	karn:ksc:cdn::account-id:domain/domain-id
CDN类型	cdn-type	karn:ksc:cdn::account-id:cdn-type/cdn-type-name
网络ACL规则	network-acl-entry	karn:ksc:vpc:region:account-id:network-acl-entry/network-acl-entry-id
安全组规则	security-group-entry	karn:ksc:vpc:region:account-id:security-group-entry/security-group-entry-id
存储空间	bucket	karn:ksc:ks3::bucket
对象(文件)	object	karn:ksc:ks3::bucket/objectKey(详细示例请参见： https://docs.ksyun.com/documents/5178)