# 云市场授权中心文档

## 1.概述

软件授权中心是阿里云基于云上软件部署及售卖提供的一种安全便捷的软件开通及生命周期的控制的方式，用户在购买ISV的软件后，云市场会为每个购买的应用生产一个软件授权码，用户在使用软件时，会输入授权码，ISV校验授权码的有效性，并为用户激活相关的应用。

接入软件授权中心过程和用户激活流程：



## 2. ISV主账号创建访问控制子账号

### 2.1主账号使用访问控制 RAM

为保障ISV的阿里云主账号安全，云市场授权码API要求使用RAM用户AK调用，RAM使用说明：[**ram简介[http://docs.alibaba-inc.com:8090/images/icons/linkext7.gif](https://help.aliyun.com/document_detail/28627.html)**](https://help.aliyun.com/document_detail/28627.html)

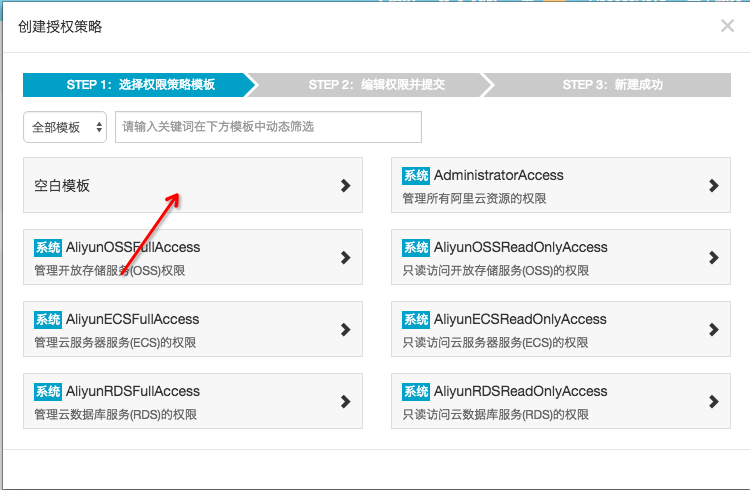
#### 2.1.1 在阿里云控制台进入[访问控制](https://ram.console.aliyun.com/)页面：

#### 2.1.2 在访问控制》授权策略管理中新建一个授权策略，如下图所示：

第一步：打开“访问控制 RAM”控制台，点击右上角“新建授权策略”；



第二步：选择空白模版；



第三步：填写授权策略名称“AliyunMarketFullAccess”，填写策略内容，如下面代码区所示内容，点击右下角“新建授权策略”按钮；

|  |
| --- |
| {    "Version": "1",    "Statement": [      {        "Effect": "Allow",        "Action": "Market:\*",        "Resource": "\*"      }    ]  } |



第四步：创建成功；

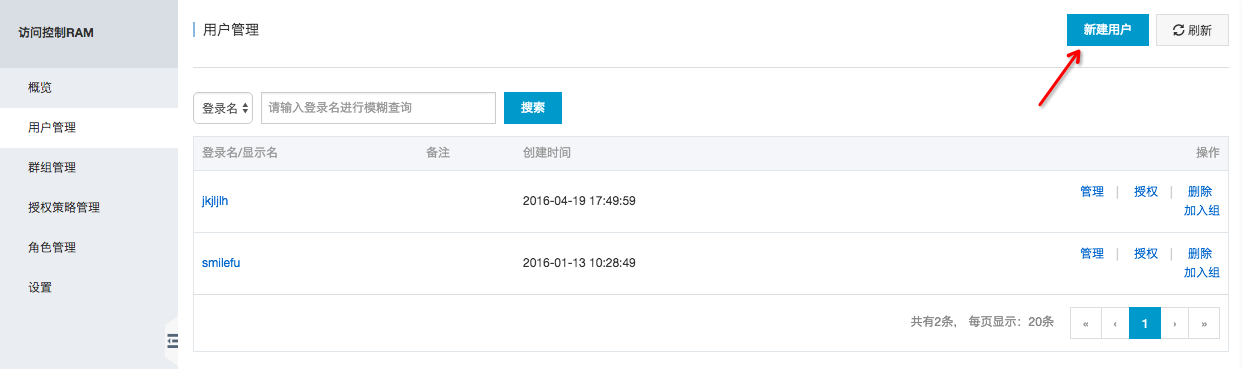


第五步：在自定义授权策略中查看刚刚创建的授权策略；

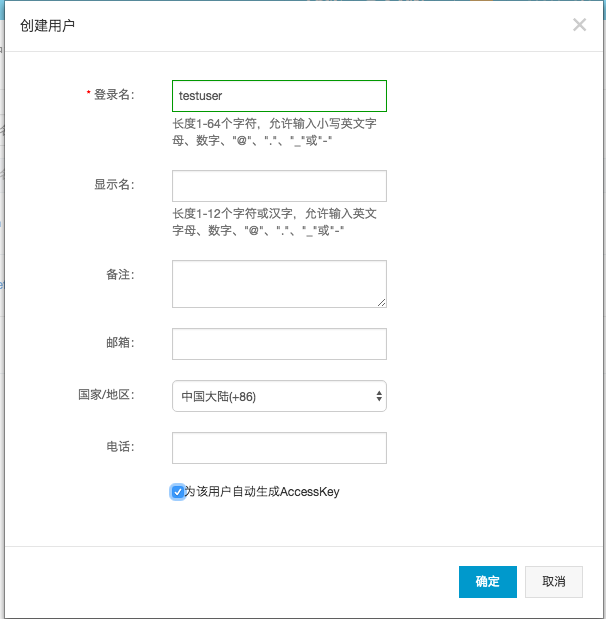
****

#### 2.1.3 在访问控制RAM 》用户管理中新建一个用户；

第一步：打开“访问控制RAM”控制台中切换到“用户管理”页面，点击右上角新建用户



第二步：创建用户，并选择“为该用户自动生成AccessKey”，点击“确定”；



第三步：点击“获取手机验证码”按钮并输入手机验证码；

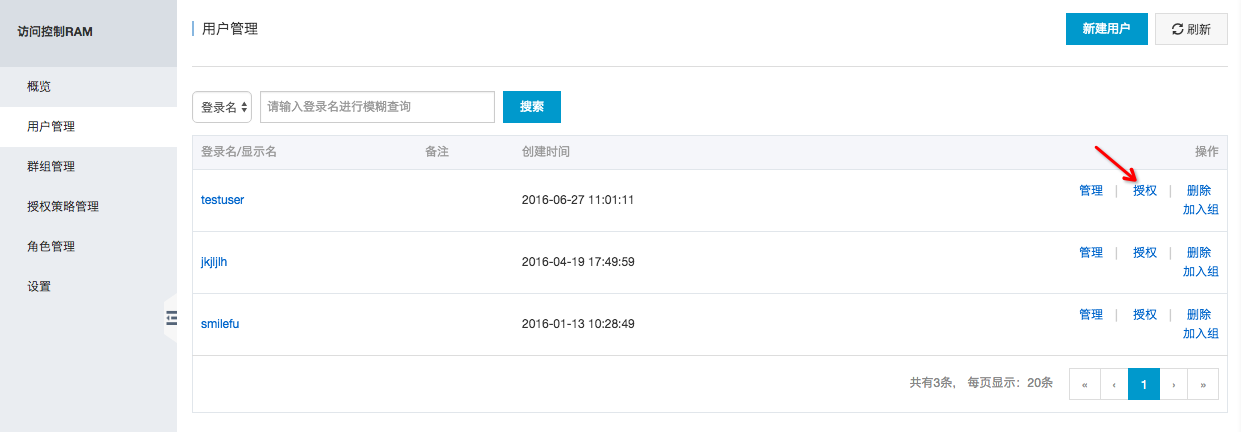


第四步：保存用户的AK信息；

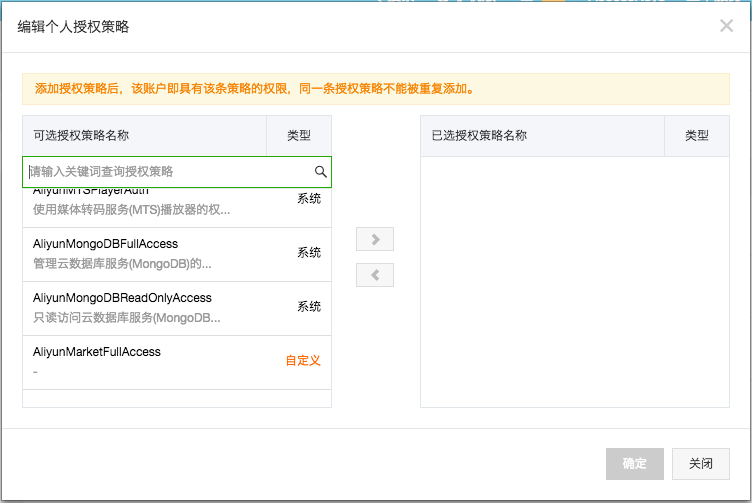
****

#### 2.1.4 为2.1.3中的新建用户进行授权

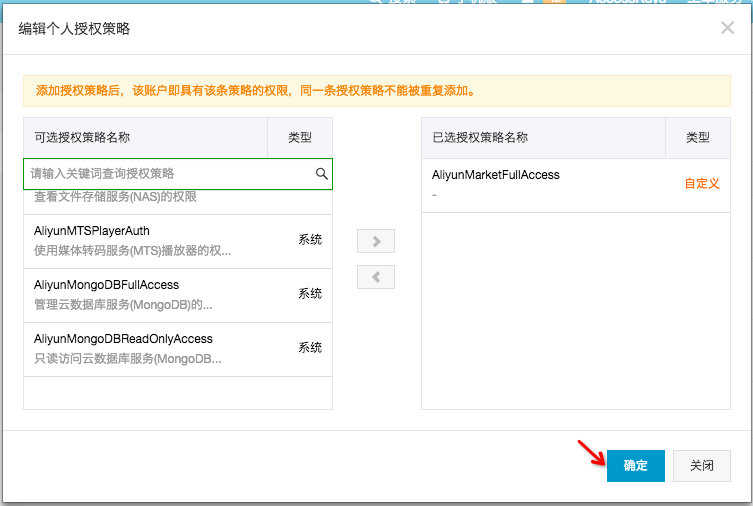
第一步：在用户管理，找到刚刚创建的用户“testuser”，点击右侧的授权按钮；



第二步：输入授权策略名称，活着滑动到最下方查看到刚才创建的授权策略；



第三步：选择授权策略，并点击右下角的“确定”按钮，为该用户添加的授权策略就成功了。



## 3. 授权地址

ISV在发布商品过程中，必须填写授权激活地址。

## 4. 接口调用公共信息文档

### 4.1 接口公共调用内容

#### 4.1.1调用约束内容：

#### 接口地址：market.aliyuncs.com

#### 通信协议：使用 https 通道发送请求

#### 字符编码：UTF-8

#### 4.1.2 公共参数：公共请求参数是指每个接口都需要使用到的请求参数

|  |  |  |  |
| --- | --- | --- | --- |
| **名称** | **类型** | **是否必须** | **描述** |
| Format | String | 否 | 返回值的类型，支持 JSON 与 XML。默认为 XML。 |
| Version | String | 是 | API 版本号，为日期形式：YYYY-MM-DD，本版本对应为 2015-11-01。 |
| AccessKeyId | String | 是 | 阿里云颁发给用户的访问服务所用的密钥 ID，云市场授权码服务要求使用子帐号ID并授权。 |
| Signature | String | 是 | 签名结果串，关于签名的计算方法，请参见<[签名机制[http://docs.alibaba-inc.com:8090/images/icons/linkext7.gif](http://www.atatech.org/articles/39913%239)](https://help.aliyun.com/document_detail/25492.html?spm=5176.product8314827_25365.6.225.QGrQQD)>。 |
| SignatureMethod | String | 是 | 签名方式，目前支持 HMAC-SHA1。 |
| Timestamp | String | 是 | 请求的时间戳。日期格式按照 [ISO8601[http://docs.alibaba-inc.com:8090/images/icons/linkext7.gif](https://help.aliyun.com/document_detail/ecs/open-api/appendix/DateTimeFormat.html)](https://help.aliyun.com/document_detail/ecs/open-api/appendix/DateTimeFormat.html) 标准表示，并需要使用 UTC 时间。格式为：  YYYY-MM-DDThh:mm:ssZ  例如，2016-05-26T12:00:00Z（为北京时间 2016 年 5 月 26 日 20 点 0 分 0 秒）。 |
| SignatureVersion | String | 是 | 签名算法版本，目前版本是 1.0。 |
| SignatureNonce | String | 是 | 唯一随机数，用于防止网络重放攻击。同一个用户，15分钟内，SignatureNonce只能使用一次。 |

### 4.2 请求示例:

[https://market.aliyuncs.com/[http://docs.alibaba-inc.com:8090/images/icons/linkext7.gif](https://market.aliyuncs.com/)](https://market.aliyuncs.com/)  
?Format=xml  
&Version=2015-11-01  
&Signature=Pc5WB8gokVn0xfeu%2FZV%2BiNM1dgI%3D  
&SignatureMethod=HMAC-SHA1  
&SignatureNonce=15215528852396  
&SignatureVersion=1.0  
&AccessKeyId=key-test  
&Timestamp=2016-06-06T12:00:00Z  
…

### 4.3公共返回参数

用户发送的每次接口调用请求，系统会返回一个唯一识别码 RequestId 给用户。

**示例XML示例：**

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

JSON示例：

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

### 4.4 公共错误码

|  |  |  |  |
| --- | --- | --- | --- |
| **code** | **message** | **httpcode** | **描述** |
| InvalidAccessKeyId.NotFound | The Access Key ID provided does not exist in our records. | 400 | 无效的 AccessKeyId 值（该 key 不存在） |
| IncompleteSignature | The request signature does not conform to Aliyun standards. | 400 | 无效的 Signature 取值（签名结果错误） |
| UnsupportedParameter | The parameter ”<parameter name>” is not supported. | 400 | 使用了无效的参数 |
| MissingParameter | The input parameter "Action" that is mandatory for processing this request is not supplied | 400 | 缺少 Action 字段 |
| ServiceUnavailable | The request has failed due to a temporary failure of the server. | 500 | 服务不可用 |
| InternalError | The request processing has failed due to some unknown error, exception or failure. | 500 | 其他情况 |

## 5. 接口调用

### 5.1 查询授权信息

**描述**  
根据授权码查询业务信息

* 首先验证授权码是否有效，无效授权码会直接返回错误信息；
* 有效授权码信息，则返回授权信息

**请求参数**

|  |  |  |  |
| --- | --- | --- | --- |
| **名称** | **类型** | **是否必须** | **描述** |
| Action | String | 是 | 系统规定参数，取值：DescribeLicense |
| LicenseCode | String | 是 | 授权码 |

**返回参数**

|  |  |  |  |
| --- | --- | --- | --- |
| **名称** | **类型** | **是否必有** | **描述** |
| InstanceId | String | 是 | 云市场业务的实例ID |
| ProductCode | String | 是 | 商品code |
| ProductName | String | 是 | 商品名称 |
| ProductSkuId | String | 是 | 商品规格标示 |
| LicenseCode | String | 是 | 授权码 |
| ExpiredTime | String | 是 | 业务到期时间，按照ISO8601标准表示，并需要使用UTC时间。格式为：YYYY-MM-DDThh:mmZ |
| LicenseStatus | String | 是 | 授权状态 Activated已激活 Inactivated 未激活 Invalid无效 |
| CreateTime | String | 是 | 创建时间，按照ISO8601标准表示，并需要使用UTC时间。格式为：YYYY-MM-DDThh:mmZ |
| ActivateTime | String | 否 | 激活时间，按照ISO8601标准表示，并需要使用UTC时间。格式为：YYYY-MM-DDThh:mmZ |
| ExtendInfo | ExtendInfo | 是 | 扩展信息 |

**扩展信息------ExtendInfo**

|  |  |  |  |
| --- | --- | --- | --- |
| **名称** | **类型** | **是否必有** | **描述** |
| AliUid | String | 是 | 购买者唯一标示 |
| Email | String | 否 | 用户email |
| Mobile | String | 否 | 用户手机号 |
| AccountQuantity | Long | 否 | 购买帐号数量 |

**错误码**

|  |  |  |  |
| --- | --- | --- | --- |
| **code** | **message** | **httpcode** | **描述** |
| License.Invalid | Invalid License | 400 | 无效的授权码 |
| License.Discard | License Discard | 400 | 授权码已废弃 |
| Auth.ISV.Error | Isv is not found | 400 | 供应商不存在 |
| Auth.SubAccount.Error | Accesskey is not subAccount | 400 | 未使用子账户调用 |
| Auth.Authorized | SubAccount is not be authorized | 400 | 子帐号未授权 |
| Auth.Match | License is not matched isv | 400 | 授权码不属于该isv |

**返回值示例：**

|  |
| --- |
| {      "License": {          "CreateTime": "2016-05-18T14:14Z",          "ExpiredTime": "2016-06-04T00:00Z",          "InstanceId": "10001165",          "LicenseCode": "ZEJLPPNWNSC1PLMPQGSMP1FZ4ECD7KE7JCPRAAA3YJ",          "LicenseStatus": "INACTIVATED",          "ProductCode": "cmgj001111",          "ProductName": "示例商品",          "ProductSkuId": "cmgj001111-code34600",          "ExtendInfo": {              "Aliuid":"11111111",              "Email": "test@aliyun.com"          }      },      "RequestId": "A007A214-4B7D-40F9-B617-A1C0C1D49FD1"  } |

### 5.2 激活授权

**描述**  
根据授权码激活业务

* 首先验证授权码是否有效，无效授权码或已激活会直接返回错误信息；

**请求参数**

|  |  |  |  |
| --- | --- | --- | --- |
| **名称** | **类型** | **是否必须** | **描述** |
| Action | String | 是 | 系统规定参数，取值：ActivateLicense |
| LicenseCode | String | 是 | 授权码 |
| Identification | String | 是 | 激活的身份信息 |

**返回参数**

|  |  |  |  |
| --- | --- | --- | --- |
| **名称** | **类型** | **是否必有** | **描述** |
| Success | bool | 是 | 成功状态标识 |

**错误码**

|  |  |  |  |
| --- | --- | --- | --- |
| **code** | **message** | **httpcode** | **描述** |
| License.Invalid | Invalid License | 400 | 无效的授权码 |
| License.Discard | License Discard | 400 | 授权码已废弃 |
| License.Expired | License Expired | 400 | 授权码已过期 |
| Auth.ISV | Isv is not found | 400 | 供应商不存在 |
| Auth.SubAcount | Accesskey is not subAcount | 400 | 未使用子账户调用 |
| Auth.Authorized | SubAcount is not be authorized | 400 | 子帐号未授权 |
| Auth.Match | License is not matched isv | 400 | 授权码不属于该isv |

**返回值示例：**

|  |
| --- |
| {  "RequestId":   "6EF60BEC-0242-43AF-BB20-270359FB54A7",  "Success":   "true"  } |

## 6. 服务商使用授权码场景

非斜体字描述了用户小王在使用软件的过程，斜体字部分是服务商在对应用户步骤应该进行的动作。

（1）小王5月6日在云市场买了1个月的易客CRM软件，在云市场管理控制台看到了易客CRM软件的授权码激活地址和授权码

（2）小王打开授权码激活地址

（3）小王在网页的页面中输入了授权码

*服务商进行操作：首先调用授权码查询接口，查询授权码信息是否有效；如果有效再调用授权码激活接口将授权码进行激活*

（4）小王觉得易客CRM软件挺好用的，于是在5月20日在云市场对易客CRM续费了3个月

*服务商进行操作：服务商要设计一个登录校验授权的机制，即用户在每次登录的时候都要调用授权码查询接口检查授权码，这样在小王下次使用软件的时候就能够及时更新用户相关信息*

（5）小王的软件在9月6日到期了，小王没有在云市场进行续费

*服务商进行操作：小王9月6日登录时，服务商调用授权码查询接口，查询到小王的使用时间过期了，那么在系统中提示小王到云市场进行续费操作，并对相关功能做禁用*

（6）小王把易客CRM软件推荐给朋友小李，小李在使用了三四天后，觉得并不适合他，于是小李发起了5天无理由退款，退款成功后没有登录过易客CRM

*服务商进行操作：每天定时查询所有授权码用户在授权中心的状态，并将状态及时更新到本地数据库，这样就能够避免资源闲置，保证本地数据和授权中心数据一致性*