

阿里云 访问控制

SDK参考

文档版本：20190906

法律声明

阿里云提醒您在使用或阅读本文档之前仔细阅读、充分理解本法律声明各条款的内容。如果您阅读或使用本文档，您的阅读或使用行为将被视为对本声明全部内容的认可。

1. 您应当通过阿里云网站或阿里云提供的其他授权通道下载、获取本文档，且仅能用于自身的合法合规的业务活动。本文档的内容视为阿里云的保密信息，您应当严格遵守保密义务；未经阿里云事先书面同意，您不得向任何第三方披露本手册内容或提供给任何第三方使用。
2. 未经阿里云事先书面许可，任何单位、公司或个人不得擅自摘抄、翻译、复制本文档内容的部分或全部，不得以任何方式或途径进行传播和宣传。
3. 由于产品版本升级、调整或其他原因，本文档内容有可能变更。阿里云保留在没有任何通知或者提示下对本文档的内容进行修改的权利，并在阿里云授权通道中不时发布更新后的用户文档。您应当实时关注用户文档的版本变更并通过阿里云授权渠道下载、获取最新版的用户文档。
4. 本文档仅作为用户使用阿里云产品及服务的参考性指引，阿里云以产品及服务的”现状“、“有缺陷”和“当前功能”的状态提供本文档。阿里云在现有技术的基础上尽最大努力提供相应的介绍及操作指引，但阿里云在此明确声明对本文档内容的准确性、完整性、适用性、可靠性等不作任何明示或暗示的保证。任何单位、公司或个人因为下载、使用或信赖本文档而发生任何差错或经济损失的，阿里云不承担任何法律责任。在任何情况下，阿里云均不对任何间接性、后果性、惩戒性、偶然性、特殊性或刑罚性的损害，包括用户使用或信赖本文档而遭受的利润损失，承担责任（即使阿里云已被告知该等损失的可能性）。
5. 阿里云网站上所有内容，包括但不限于著作、产品、图片、档案、资讯、资料、网站架构、网站画面的安排、网页设计，均由阿里云和/或其关联公司依法拥有其知识产权，包括但不限于商标权、专利权、著作权、商业秘密等。非经阿里云和/或其关联公司书面同意，任何人不得擅自使用、修改、复制、公开传播、改变、散布、发行或公开发表阿里云网站、产品程序或内容。此外，未经阿里云事先书面同意，任何人不得为了任何营销、广告、促销或其他目的使用、公布或复制阿里云的名称（包括但不限于单独为或以组合形式包含”阿里云”、Aliyun”、“万网”等阿里云和/或其关联公司品牌，上述品牌的附属标志及图案或任何类似公司名称、商号、商标、产品或服务名称、域名、图案标示、标志、标识或通过特定描述使第三方能够识别阿里云和/或其关联公司）。
6. 如若发现本文档存在任何错误，请与阿里云取得直接联系。

通用约定

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

目录

法律声明.....	I
通用约定.....	I
1 SDK参考 (RAM)	1
1.1 RAM SDK概览.....	1
1.2 Java SDK.....	1
1.3 .NET SDK.....	3
1.4 Python SDK.....	4
1.5 PHP SDK.....	5
1.6 Node.js SDK.....	6
1.7 (隐藏) 旧版 SDK 参考 (RAM)	6
1.7.1 Java SDK.....	6
1.7.1.1 Java SDK 示例.....	6
1.7.2 .NET SDK.....	7
1.7.2.1 .NET SDK 示例.....	7
1.7.3 Python SDK.....	8
1.7.3.1 Python SDK 示例.....	8
1.7.4 PHP SDK.....	9
1.7.4.1 PHP SDK 示例.....	9
1.7.5 Node.js SDK.....	9
1.7.5.1 Node.js SDK 示例.....	9
2 SDK参考 (STS)	10
2.1 STS SDK概览.....	10
2.2 Java SDK.....	10
2.3 .NET SDK.....	12
2.4 Python SDK.....	14
2.5 PHP SDK.....	14
2.6 Node.js SDK.....	16

1 SDK参考 (RAM)

1.1 RAM SDK概览

本文主要介绍RAM SDK的使用，在使用SDK前您需要了解RAM API，熟悉各个接口的具体功能。

RAM SDK支持的语言

- [#unique_5](#)
- [#unique_6](#)
- [#unique_7](#)
- [#unique_8](#)
- [#unique_9](#)

1.2 Java SDK

本文简要介绍了Java SDK的安装并提供一个简单的操作示例。

背景信息

RAM SDK包含阿里云Java SDK公共部分和RAM部分：

- 阿里云Java SDK公共部分依赖aliyun-java-sdk-core。关于SDK示例代码的自动生成和在线API调试，请参见[OpenAPI Explorer](#)。
- RAM部分依赖aliyun-java-sdk-ram。关于RAM API接口相关信息，请参见[#unique_11](#)。

Java SDK的安装

您可以通过Maven管理项目依赖或手动下载RAM SDK的jar包后手动添加到项目中。

- 通过Maven管理项目依赖（推荐）。

1. 使用Maven创建项目。

```
mvn archetype:generate -DgroupId=com.aliyun.ram.sample \
-DartifactId=ram-sdk-sample \
-Dpackage=com.aliyun.ram.sample \
```

```
-Dversion=1.0-SNAPSHOT
```

2. 在项目的pom.xml文件中加入相应依赖项。

添加aliyun-java-sdk的相关依赖，以2.0.7版本为例，在标签内添加如下内容：

```
<dependency>
  <groupId>com.aliyun</groupId>
  <artifactId>aliyun-java-sdk-ram</artifactId>
  <version>2.0.7</version>
</dependency>
<dependency>
  <groupId>com.aliyun</groupId>
  <artifactId>aliyun-java-sdk-core</artifactId>
  <version>[4.4.2,5.0.0)</version>
</dependency>
```



说明：

aliyun-java-sdk已经加入到Maven仓库中，无需设置settings.xml。

- 手动下载RAM SDK的jar包后手动添加到项目中。

下载地址：

- [Java SDK](#)
- [RAM SDK](#)

Java SDK示例

以下示例以Java SDK为例，说明如何创建RAM用户。

```
package com.aliyun.ram.sample;

import com.aliyuncs.DefaultAcsClient;
import com.aliyuncs.exceptions.ClientException;
import com.aliyuncs.profile.DefaultProfile;
import com.aliyuncs.profile.IClientProfile;

// 当前RAM API版本为：2015-05-01
import com.aliyuncs.ram.model.v20150501.*;

/**
 * Created by JasonGao on 15/11/4.
 */

public class RamServiceSample {

    public static void main(String[] args) {
        // 构建一个阿里云client，用于发起请求
        // 构建阿里云client时需要设置AccessKey ID和AccessKey Secret
        // RAM是Global Service，API入口位于华东 1（杭州），这里Region填写：
        cn-hangzhou
        IClientProfile profile = DefaultProfile.getProfile("cn-
        hangzhou",
        "AccessKeyId">
```

```
AccessKeySecret>");
    DefaultAcsClient client = new DefaultAcsClient(profile);

    // 构建CreateUser请求
    final CreateUserRequest request = new CreateUserRequest();

    //设置参数: UserName
    request.setUserName("alice");

    // 发起请求, 并得到response
    try {
        final CreateUserResponse response = client.getAcsResponse(
request);

        System.out.println("UserName: " + response.getUser().
getUserName());
        System.out.println("CreateTime: " + response.getUser().
getCreateDate());
    } catch (ClientException e) {
        System.out.println("Failed.");
        System.out.println("Error code: " + e.getErrCode());
        System.out.println("Error message: " + e.getErrMsg());
    }
}
}
```

1.3 .NET SDK

本文简要介绍了.NET SDK的安装并提供一个简单的操作示例。

.NET SDK的安装

安装包下载地址：[.NET SDK](#)。



说明:

RAM已经接入NuGet, 详情请参见[RAM NuGet](#)。

.NET SDK示例

以下示例以 .NET SDK为例, 说明如何创建RAM用户。

```
using System;
using Aliyun.Acs.Core;
using Aliyun.Acs.Core.Profile;
using Aliyun.Acs.Ram.Model.V20150501;

namespace ram_net_sdk_sample
{
    class Program
    {
        static void Main(string[] args)
        {
            // 构建一个阿里云client, 用于发起请求
            // 构建阿里云client时需要设置AccessKey ID和AccessKey Secret
```

```
// RAM是Global Service, API入口位于华东 1 (杭州), 这里Region
填写:cn-hangzhou
IClientProfile profile = DefaultProfile.GetProfile("cn-
hangzhou", "<access-key-id>", "<access-key-secret>");
DefaultAcsClient client = new DefaultAcsClient(profile);

// 构造CreateUser请求
CreateUserRequest request = new CreateUserRequest();

//设置参数: UserName
request.UserName = "alice";

// 发起请求, 并得到response
try
{
    CreateUserResponse response = client.GetAcsResponse(
request);

    Console.WriteLine("UserName: " + response.User.
UserName);
    Console.WriteLine("CreateTime: " + response.User.
CreateDate);
}
catch (Exception ex)
{
    Console.WriteLine(ex.ToString());
}
Console.ReadLine();
}
}
```

1.4 Python SDK

本文简要介绍了Python SDK的安装并提供一个简单的操作示例。

Python SDK的安装

您可以使用pip或手动下载Python安装包后手动添加到项目中。

- 通过pip安装: `pip install aliyun-python-sdk-ram`。
- Python SDK安装包下载地址:
 - [Python SDK](#)
 - [RAM SDK](#)

Python SDK示例

以下示例以Python SDK为例, 说明如何创建RAM用户。

```
#!/usr/bin/env python
#coding=utf-8

from aliyunsdkcore import client
from aliyunsdkram.request.v20150501 import CreateUserRequest
```



```
# 构建一个阿里云client, 用于发起请求
# 构建阿里云client时需要设置AccessKey ID和AccessKey Secret
# RAM是Global Service, API入口位于华东 1 (杭州), 这里Region填写:cn-
hangzhou
clt = client.AcsClient('<access-key-id>', '<access-key-secret>', 'cn-
hangzhou')

# 构造CreateUser请求
request = CreateUserRequest.CreateUserRequest()
# 设置参数: UserName
request.set_UserName('alice')

# 发起请求, 并得到response
response = clt.do_action(request)

print response
```

1.5 PHP SDK

本文简要介绍了PHP SDK的安装并提供一个简单的操作示例。

PHP SDK的安装

安装包下载地址：[PHP SDK](#)。

PHP SDK示例

以下示例以PHP SDK为例，说明如何创建RAM用户。

```
<?php
include_once 'aliyun-php-sdk-core/Config.php';
use Ram\Request\V20150501 as Ram;

// 构建一个阿里云client, 用于发起请求
// 构建阿里云client时需要设置AccessKey ID和AccessKey Secret
// RAM是Global Service, API入口位于华东 1 (杭州), 这里Region填写:cn-
hangzhou
$iClientProfile = DefaultProfile::getProfile("cn-hangzhou", "<access-
key-id>", "<access-key-secret>");
$client = new DefaultAcsClient($iClientProfile);

// 构造CreateUser请求
$request = new Ram\CreateUserRequest();
//设置参数: UserName
$request->setUserName("alice");

//发起请求, 并得到response
$response = $client->doAction($request);

print_r("\r\n");
print_r($response);
```

```
?>
```

1.6 Node.js SDK

本文简要介绍了Node.js SDK的安装并提供一个简单的操作示例。

Node.js SDK的安装

您可以通过npm安装Node.js SDK: `npm install aliyun-sdk`。

Node.js SDK示例

以下示例以Node.js SDK为例, 说明如何创建RAM用户。

```
ALY = require("aliyun-sdk")

// 构建一个阿里云client, 用于发起请求
// 构建阿里云client时需要设置AccessKey ID和AccessKey Secret
// RAM是Global Service, API入口位于华东 1 (杭州), 这里使用RAM API的主地址
var ram = new ALY.RAM({
    accessKeyId: "<access-key-id>",
    AccessKeysecret: "<access-key-secret>",
    endpoint: 'https://ram.aliyuncs.com',
    apiVersion: '2015-05-01'
});

// 构造CreateUser请求
ram.createUser({
    Action: 'CreateUser',
    UserName: "alice"
}, function (err, res) {
    console.log(err, res);
});
```

1.7 (隐藏) 旧版 SDK 参考 (RAM)

1.7.1 Java SDK

1.7.1.1 Java SDK 示例

本文以Java SDK为例, 说明如何创建RAM用户。

```
package com.aliyun.ram.sample;

import com.aliyuncs.DefaultAcsClient;
import com.aliyuncs.exceptions.ClientException;
import com.aliyuncs.profile.DefaultProfile;
import com.aliyuncs.profile.IClientProfile;

// 当前 RAM API 版本为: 2015-05-01
import com.aliyuncs.ram.model.v20150501.*;

/**
 * Created by JasonGao on 15/11/4.
 */
```

```
public class RamServiceSample {

    public static void main(String[] args) {
        // 构建一个阿里云 Client, 用于发起请求
        // 构建阿里云 Client 时需要设置 access key ID 和 access key secret
        // RAM 是 Global Service, API 入口位于华东 1 (杭州), 这里 Region
        // 填写:cn-hangzhou
        IClientProfile profile = DefaultProfile.getProfile("cn-
hangzhou",
                                                    "<
AccessKeyId>",
                                                    "<
AccessKeySecret>");
        DefaultAcsClient client = new DefaultAcsClient(profile);

        // 构建 CreateUser 请求
        final CreateUserRequest request = new CreateUserRequest();

        //设置参数: UserName
        request.setUserName("alice");

        // 发起请求, 并得到 response
        try {
            final CreateUserResponse response = client.getAcsResponse(
request);

            System.out.println("UserName: " + response.getUser().
getUserName());
            System.out.println("CreateTime: " + response.getUser().
getCreateDate());
        } catch (ClientException e) {
            System.out.println("Failed.");
            System.out.println("Error code: " + e.getErrCode());
            System.out.println("Error message: " + e.getErrMsg());
        }
    }
}
```

1.7.2 .NET SDK

1.7.2.1 .NET SDK 示例

本文以 .NET SDK 为例, 说明如何创建 RAM 用户。

```
using System;
using Aliyun.Acs.Core;
using Aliyun.Acs.Core.Profile;
using Aliyun.Acs.Ram.Model.V20150501;

namespace ram_net_sdk_sample
{
    class Program
    {
        static void Main(string[] args)
        {
            // 构建一个阿里云 Client, 用于发起请求
            // 构建阿里云 Client 时需要设置 access key ID 和 access key
            // secret
            // RAM 是 Global Service, API 入口位于华东 1 (杭州), 这里
            // Region 填写:cn-hangzhou
```

```
IClientProfile profile = DefaultProfile.GetProfile("cn-
hangzhou", "<access-key-id>", "<access-key-secret>");
DefaultAcsClient client = new DefaultAcsClient(profile);

// 构造 CreateUser 请求
CreateUserRequest request = new CreateUserRequest();

//设置参数: UserName
request.UserName = "alice";

// 发起请求, 并得到 response
try
{
    CreateUserResponse response = client.GetAcsResponse(
request);

    Console.WriteLine("UserName: " + response.User.
UserName);
    Console.WriteLine("CreateTime: " + response.User.
CreateDate);
}
catch (Exception ex)
{
    Console.WriteLine(ex.ToString());
}
Console.ReadLine();
}
}
```

1.7.3 Python SDK

1.7.3.1 Python SDK 示例

本文以 Python SDK 为例, 说明如何创建 RAM 用户。

```
#!/usr/bin/env python
#coding=utf-8

from aliyunsdkcore import client
from aliyunsdkram.request.v20150501 import CreateUserRequest

# 构建一个阿里云 Client, 用于发起请求
# 构建阿里云 Client 时需要设置 access key ID 和 access key secret
# RAM 是 Global Service, API 入口位于华东 1 (杭州), 这里 Region 填写:cn-
hangzhou
clt = client.AcsClient('<access-key-id>', '<access-key-secret>', 'cn-
hangzhou')

# 构造 CreateUser 请求
request = CreateUserRequest.CreateUserRequest()
# 设置参数: UserName
request.set_UserName('alice')

# 发起请求, 并得到 response
response = clt.do_action(request)
```

```
print response
```

1.7.4 PHP SDK

1.7.4.1 PHP SDK 示例

本文以 PHP SDK 为例，说明如何创建 RAM 用户。

```
<?php
include_once 'aliyun-php-sdk-core/Config.php';
use Ram\Request\V20150501 as Ram;

// 构建一个阿里云 Client，用于发起请求
// 构建阿里云 Client 时需要设置 access key ID 和 access key secret
// RAM 是 Global Service，API 入口位于华东 1（杭州），这里 Region 填写:cn-
hangzhou
$iClientProfile = DefaultProfile::getProfile("cn-hangzhou", "<access-
key-id>", "<access-key-secret>");
$client = new DefaultAcsClient($iClientProfile);

// 构造 CreateUser 请求
$request = new Ram\CreateUserRequest();
//设置参数: UserName
$request->setUserName("alice");

//发起请求，并得到 response
$response = $client->doAction($request);

print_r("\r\n");
print_r($response);
?>
```

1.7.5 Node.js SDK

1.7.5.1 Node.js SDK 示例

本文以 Node.js SDK 为例，说明如何创建 RAM 用户。

```
ALY = require("aliyun-sdk")

// 构建一个阿里云 Client，用于发起请求
// 构建阿里云 Client 时需要设置 access key ID 和 access key secret
// RAM 是 Global Service，API 入口位于华东 1（杭州），这里使用 RAM API 的主
地址
var ram = new ALY.RAM({
  accessKeyId: "<access-key-id>",
  secretAccessKey: "<access-key-secret>",
  endpoint: 'https://ram.aliyuncs.com',
  apiVersion: '2015-05-01'
});

// 构造 CreateUser 请求
ram.createUser({
  Action: 'CreateUser',
  UserName: "alice"
}, function (err, res) {
  console.log(err, res);
});
```

2 SDK参考 (STS)

2.1 STS SDK概览

本文主要介绍STS SDK的使用，在使用SDK前您需要了解STS API，熟悉各个接口的具体功能。

STS SDK支持的语言

- [Java SDK](#)
- [.NET SDK](#)
- [Python SDK](#)
- [PHP SDK](#)
- [Node.js SDK](#)

2.2 Java SDK

本文简要介绍了Java SDK的安装并提供一个简单的操作示例。

背景信息

STS SDK包含阿里云Java SDK公共部分和STS部分：

- 阿里云Java SDK公共部分依赖aliyun-java-sdk-core。关于SDK示例代码的自动生成和在线API调试，请参见[OpenAPI Explorer](#)。
- STS部分依赖aliyun-java-sdk-sts。关于STS API接口相关信息，请参见[#unique_35](#)。

STS Java SDK的安装

您可以通过Maven管理项目依赖或手动下载STS SDK的jar包后添加到项目中。

- 通过Maven管理项目依赖（推荐）。

1. 使用Maven创建项目。

```
mvn archetype:generate -DgroupId=com.aliyun.sts.sample \
-DartifactId=sts-sdk-sample \
-Dpackage=com.aliyun.sts.sample \
-Dversion=1.0-SNAPSHOT
```

2. 在项目的pom.xml文件中加入相应依赖项。

添加aliyun-java-sdk的相关依赖，以3.0.0版本为例，在标签内添加如下内容：

```
<dependency>
  <groupId>com.aliyun</groupId>
```

```
<artifactId>aliyun-java-sdk-sts</artifactId>
<version>3.0.0</version>
</dependency>
<dependency>
  <groupId>com.aliyun</groupId>
  <artifactId>aliyun-java-sdk-core</artifactId>
  <version>[4.4.2,5.0.0)</version>
</dependency>
```



说明:

aliyun-java-sdk已经加入到Maven仓库中, 无需设置settings.xml。

- 手动下载STS SDK的jar包后手动添加到项目中。

下载地址:

- [Java SDK](#)
- [GitHub](#)

Java SDK示例

在com/aliyun/sts/sample/目录下创建Java源代码StsServiceSample.java。



说明:

以下示例core以4.4.2版本为例, 仅供参考。

```
package com.aliyun.sts.sample;
import com.aliyuncs.DefaultAcsClient;
import com.aliyuncs.exceptions.ClientException;
import com.aliyuncs.http.MethodType;
import com.aliyuncs.profile.DefaultProfile;
import com.aliyuncs.profile.IClientProfile;
import com.aliyuncs.sts.model.v20150401.AssumeRoleRequest;
import com.aliyuncs.sts.model.v20150401.AssumeRoleResponse;
public class StsServiceSample {
    public static void main(String[] args) {
        String endpoint = "sts.aliyuncs.com";
        String accessKeyId = "<access-key-id>";
        String accessKeySecret = "<access-key-secret>";
        String roleArn = "<role-arn>";
        String roleSessionName = "session-name";
        String policy = "{\n" +
            "    \"Version\": \"1\", \n" +
            "    \"Statement\": [\n" +
            "        {\n" +
            "            \"Action\": [\n" +
            "                \"oss:*\"\n" +
            "            ], \n" +
            "            \"Resource\": [\n" +
            "                \"acs:oss:*:*:*\" \n" +
            "            ], \n" +
            "            \"Effect\": \"Allow\"\n" +
            "        }\n" +
            "    ]\n" +
            "    }";
        try {
```

```
// 构造default profile (参数留空, 无需添加region ID)
IClientProfile profile = DefaultProfile.getProfile("",
accessKeyId, accessKeySecret);
// 用profile构造client
DefaultAcsClient client = new DefaultAcsClient(profile);
final AssumeRoleRequest request = new AssumeRoleRequest();
request.setSysEndpoint(endpoint);
request.setSysMethod(MethodType.POST);
request.setRoleArn(roleArn);
request.setRoleSessionName(roleSessionName);
request.setPolicy(policy); // Optional
final AssumeRoleResponse response = client.getAcsResponse(
request);
    System.out.println("Expiration: " + response.getCredent
ials().getExpiration());
    System.out.println("Access Key Id: " + response.getCredent
ials().getAccessKeyId());
    System.out.println("Access Key Secret: " + response.
getCredentials().getAccessKeySecret());
    System.out.println("Security Token: " + response.
getCredentials().getSecurityToken());
    System.out.println("RequestId: " + response.getRequestId
());
    } catch (ClientException e) {
        System.out.println("Failed: ");
        System.out.println("Error code: " + e.getErrCode());
        System.out.println("Error message: " + e.getErrMsg());
        System.out.println("RequestId: " + e.getRequestId());
    }
}
```



说明:

- 请修改AccessKey ID和AccessKey Secret为有效值。
- STS各区域的endpoint, 请参见[#unique_36](#)。
- AssumeRole接口相关信息, 请参见[#unique_37](#)。

2.3 .NET SDK

本文简要介绍了.NET SDK的安装并提供一个简单的操作示例。

.NET SDK的安装

安装包下载地址:

- [.NET SDK](#)
- [GitHub](#)



说明:

STS已经接入NuGet, 详情请参见[STS NuGet](#)。

.NET SDK示例

```
using System;
using Aliyun.Acs.Core;
using Aliyun.Acs.Core.Profile;
using Aliyun.Acs.Core.Http;
using Aliyun.Acs.Sts.Model.V20150401;
namespace StsNetSdkDemo
{
    class Program
    {
        static void Main(string[] args)
        {
            const string REGIONID = "cn-shanghai";
            const string ENDPOINT = "sts.cn-shanghai.aliyuncs.com";
            // 构建一个阿里云client, 用于发起请求
            // 构建阿里云client时需要设置AccessKey ID和AccessKey Secret
            DefaultProfile.AddEndpoint(REGIONID, REGIONID, "Sts",
ENDPOINT);
            IClientProfile profile = DefaultProfile.GetProfile(
REGIONID, "<access-key-id>", "<access-key-secret>");
            DefaultAcsClient client = new DefaultAcsClient(profile);
            // 构建AssumeRole请求
            AssumeRoleRequest request = new AssumeRoleRequest();
            request.AcceptFormat = FormatType.JSON;
            // 指定角色ARN
            request.RoleArn = "<role-arn>";
            request.RoleSessionName = "<role-session-name>";
            // 设置token有效期, 可选参数, 默认3600秒;
            request.DurationSeconds = 3600;
            // 设置token的附加权限策略; 在获取token时, 通过额外设置一个权限策略
            // 进一步减小token的权限;
            // request.Policy="<policy-content>"
            try
            {
                AssumeRoleResponse response = client.GetAcsResponse(
request);
                Console.WriteLine("AccessKeyId: " + response.
Credentials.AccessKeyId);
                Console.WriteLine("AccessKeySecret: " + response.
Credentials.AccessKeySecret);
                Console.WriteLine("SecurityToken: " + response.
Credentials.SecurityToken);
                //Token过期时间; 服务器返回UTC时间, 这里转换成北京时间显示;
                Console.WriteLine("Expiration: " + DateTime.Parse(
response.Credentials.Expiration).ToLocalTime());
            }
            catch (Exception ex)
            {
                Console.WriteLine(ex.ToString());
            }
            Console.ReadLine();
        }
    }
}
```



说明:

- STS各区域的endpoint, 请参见[#unique_36](#)。
- AssumeRole接口相关信息, 请参见[#unique_37](#)。

2.4 Python SDK

本文简要介绍了Python SDK的安装并提供一个简单的操作示例。

Python SDK的安装

安装包下载地址：

- [Python SDK](#)
- [GitHub](#)

Python SDK示例

```
#!/usr/bin/env python
#coding=utf-8
from aliyunsdkcore import client
from aliyunsdkcore.profile import region_provider
from aliyunsdksts.request.v20150401 import AssumeRoleRequest
# 构建一个阿里云client，用于发起请求
# 构建阿里云client时需要设置AccessKey ID和AccessKey Secret
REGIONID = 'cn-shanghai'
ENDPOINT = 'sts.cn-shanghai.aliyuncs.com'
# 配置要访问的STS endpoint
region_provider.add_endpoint('Sts', REGIONID, ENDPOINT)
# 初始化Client
clt = client.AcsClient('<access-key-id>', '<access-key-secret>',
REGIONID)
# 构建AssumeRole请求
request = AssumeRoleRequest.AssumeRoleRequest()
# 指定角色ARN
request.set_RoleArn('<role-arn>')
# 设置会话名称，审计服务使用此名称区分调用者
request.set_RoleSessionName('<role-session-name>')
# 发起请求，并得到response
response = clt.do_action_with_exception(request)
print response
```



说明：

- STS各区域的endpoint，请参见[#unique_36](#)。
- AssumeRole接口相关信息，请参见[#unique_37](#)。

2.5 PHP SDK

本文简要介绍了PHP SDK的安装并提供一个简单的操作示例。

PHP SDK 的安装

安装包下载地址：

- [PHP SDK](#)
- [GitHub](#)

PHP SDK示例

```
<?php
include_once 'aliyun-php-sdk-core/Config.php';
use Sts\Request\V20150401 as Sts;
define("REGION_ID", "cn-shanghai");
define("ENDPOINT", "sts.cn-shanghai.aliyuncs.com");
// 只允许RAM用户使用角色
DefaultProfile::addEndpoint(REGION_ID, REGION_ID, "Sts", ENDPOINT);
$iClientProfile = DefaultProfile::getProfile(REGION_ID, "<access-key-id>", "<access-key-secret>");
$client = new DefaultAcsClient($iClientProfile);
// 指定角色ARN
$roleArn = "<role-arn>";
// 在扮演角色时, 添加一个权限策略, 进一步限制角色的权限
// 以下权限策略表示拥有可以读取所有OSS的只读权限
$policy=<<<<POLICY
{
    "Statement": [
        {
            "Action": [
                "oss:Get*",
                "oss:List*"
            ],
            "Effect": "Allow",
            "Resource": "*"
        }
    ],
    "Version": "1"
}
POLICY;
$request = new Sts\AssumeRoleRequest();
// RoleSessionName即临时身份的会话名称, 用于区分不同的临时身份
$request->setRoleSessionName("client_name");
$request->setRoleArn($roleArn);
$request->setPolicy($policy);
$request->setDurationSeconds(3600);
try {
    $response = $client->getAcsResponse($request);
    print_r($response);
} catch(ServerException $e) {
    print "Error: " . $e->getErrorCode() . " Message: " . $e->
getMessage() . "\n";
} catch(ClientException $e) {
    print "Error: " . $e->getErrorCode() . " Message: " . $e->
getMessage() . "\n";
}
?>
```



说明:

- STS各区域的endpoint, 请参见[#unique_36](#)。
- AssumeRole接口相关信息, 请参见[#unique_37](#)。

2.6 Node.js SDK

本文简要介绍了Node.js SDK的安装并提供一个简单的操作示例。

Node.js SDK的安装

通过npm安装Node.js SDK: `npm install -S @alicloud/pop-core`。

Node.js SDK示例

```
/*
  Usage: npm install -S @alicloud/pop-core
  powered by alinode (http://alinode.aliyun.com/)
*/

const Core = require('@alicloud/pop-core');
// 构建一个阿里云client, 用于发起请求
// 构建阿里云client时需要设置AccessKey ID和AccessKey Secret
var client = new Core({
  accessKeyId: '<accessKeyId>',
  AccessKeySecret: '<accessSecret>',
  endpoint: 'https://sts.aliyuncs.com',
  apiVersion: '2015-04-01'
});

//设置参数, 指定角色ARN
var params = {
  'RoleArn': '<role-arn>',
  'RoleSessionName': 'session-name'
};

//构建AssumeRole请求
client.request('AssumeRole', params).then((result) => {
  console.log(result);
}, (ex) => {
  console.log(ex);
})
```



说明:

- STS各区域的endpoint, 请参见[#unique_36](#)。
- AssumeRole接口相关信息, 请参见[#unique_37](#)。