

访问控制

STS SDK使用手册



STS SDK使用手册

Java SDK

前言

简介

本文档主要介绍STS Java SDK的安装和使用。本文档假设您已经开通了阿里云RAM服务，并为主账号或者RAM用户创建了Access Key ID 和Access Key Secret。

在使用STS Java SDK之前，建议认真阅读RAM用户指南，理解主账号、RAM用户、联盟用户和短期访问凭证等概念，以及STS的应用场景。

STS API 请参考STS API文档。

版本更迭

Java SDK开发包(2015-08-25) 版本号2.1.6

- 发布AssumeRole接口

安装

在Eclipse中使用JAR包

步骤如下：

- 在阿里云官方网站下载STS Java SDK
- 解压文件
- 将解压后文件夹中的文件： aliyun-java-sdk-core-<versionId>.jar 以及 aliyun-java-sdk-sts-<versionId>.jar 拷贝到你的工程文件夹中
- 在Eclipse右键工程 -> Properties -> Java Build Path -> Add JARs

- 选择你拷贝的所有JAR文件
- 经过上面几步之后，你就可以在工程中使用STS Java SDK了

其他IDE与Eclipse配置方法类似。

在Maven工程中使用SDK

添加aliyun-java-sdk的相关依赖。以2.1.6版本为例，在 <dependencies> 标签内加入如下内容：

```
<dependency>
  <groupId>com.aliyun</groupId>
  <artifactId>aliyun-java-sdk-sts</artifactId>
  <version>2.1.6</version>
</dependency>
<dependency>
  <groupId>com.aliyun</groupId>
  <artifactId>aliyun-java-sdk-core</artifactId>
  <version>2.1.7</version>
</dependency>
```

使用入门

创建和管理角色

STS的AssumeRole接口的使用，需要在RAM中创建和管理角色，请参考[RAM角色管理](#)

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
```

修改生成的pom.xml，添加aliyun-java-sdk的相关依赖。以2.1.6版本为例，在 <dependencies> 标签内加入如下内容：

```
<dependency>
  <groupId>com.aliyun</groupId>
  <artifactId>aliyun-java-sdk-sts</artifactId>
  <version>2.1.6</version>
</dependency>
<dependency>
```

```
<groupId>com.aliyun</groupId>
<artifactId>aliyun-java-sdk-core</artifactId>
<version>2.1.7</version>
</dependency>
```

aliyun-java-sdk已经加入到 <https://maven-repository.com/artifact/com.aliyun>

无需设置maven的settings.xml。

aliyun-java-sdk-sts使用的示例代码

- 注意：请修改accessKeyId和accessKeySecret为有效值。
- 在com.aliyun/sts/sample/目录下创建Java源代码StsServiceSample.java，内容如下：

```
package com.aliyun.sts.sample;

import com.aliyuncs.DefaultAcsClient;
import com.aliyuncs.exceptions.ClientException;
import com.aliyuncs.http.MethodType;
import com.aliyuncs.http.ProtocolType;
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 {
    // 目前只有"cn-hangzhou"这个region可用, 不要使用填写其他region的值
    public static final String REGION_CN_HANGZHOU = "cn-hangzhou";
    // 当前 STS API 版本
    public static final String STS_API_VERSION = "2015-04-01";

    static AssumeRoleResponse assumeRole(String accessKeyId, String accessKeySecret,
        String roleArn, String roleSessionName, String policy,
        ProtocolType protocolType) throws ClientException {
        try {
            // 创建一个 Aliyun Acs Client, 用于发起 OpenAPI 请求
            IClientProfile profile = DefaultProfile.getProfile(REGION_CN_HANGZHOU, accessKeyId, accessKeySecret);
            DefaultAcsClient client = new DefaultAcsClient(profile);

            // 创建一个 AssumeRoleRequest 并设置请求参数
            final AssumeRoleRequest request = new AssumeRoleRequest();
            request.setVersion(STS_API_VERSION);
            request.setMethod(MethodType.POST);
            request.setProtocol(protocolType);

            request.setRoleArn(roleArn);
            request.setRoleSessionName(roleSessionName);
            request.setPolicy(policy);

            // 发起请求, 并得到response
            final AssumeRoleResponse response = client.getAcsResponse(request);
```

```

        return response;
    } catch (ClientException e) {
        throw e;
    }
}

public static void main(String[] args) {
    // 只有 RAM用户 (子账号) 才能调用 AssumeRole 接口
    // 阿里云主账号的AccessKeys不能用于发起AssumeRole请求
    // 请首先在RAM控制台创建一个RAM用户, 并为这个用户创建AccessKeys
    String accessKeyId = "o*****F";
    String accessKeySecret = "y*****U";

    // AssumeRole API 请求参数: RoleArn, RoleSessionName, Polciy, and DurationSeconds

    // RoleArn 需要在 RAM 控制台上获取
    String roleArn = "acs:ram::145883****900618:role/ossadminrole";

    // RoleSessionName 是临时Token的会话名称, 自己指定用于标识你的用户, 主要用于审计, 或者用于区分Token颁发给谁
    // 但是注意RoleSessionName的长度和规则, 不要有空格, 只能有'-' '_' 字母和数字等字符
    // 具体规则请参考API文档中的格式要求
    String roleSessionName = "alice-001";

    // 如何定制你的policy?
    String policy = "{\n" +
        "  \"Version\": \"1\", \n" +
        "  \"Statement\": [\n" +
        "    {\n" +
        "      \"Action\": [\n" +
        "        \"oss:GetBucket\", \n" +
        "        \"oss:GetObject\" \n" +
        "      ], \n" +
        "      \"Resource\": [\n" +
        "        \"acs:oss:*:*\" \n" +
        "      ], \n" +
        "      \"Effect\": \"Allow\" \n" +
        "    } \n" +
        "  ] \n" +
        "}";

    // 此处必须为 HTTPS
    ProtocolType protocolType = ProtocolType.HTTPS;

    try {
        final AssumeRoleResponse response = assumeRole(accessKeyId, accessKeySecret,
            roleArn, roleSessionName, policy, protocolType);

        System.out.println("Expiration: " + response.getCredentials().getExpiration());
        System.out.println("Access Key Id: " + response.getCredentials().getAccessKeyId());
        System.out.println("Access Key Secret: " + response.getCredentials().getAccessKeySecret());
        System.out.println("Security Token: " + response.getCredentials().getSecurityToken());
    } catch (ClientException e) {
        System.out.println("Failed to get a token.");
        System.out.println("Error code: " + e.getErrCode());
        System.out.println("Error message: " + e.getErrMsg());
    }
}

```

```
}
}
```

编译运行示例代码

编译

```
mvn install
```

运行

```
mvn -q exec:java -Dexec.mainClass=com.aliyun.sts.sample.StsServiceSample
```

Java SDK 开发包

Java SDK开发包(2015-08-25) 版本号2.1.6

Java SDK下载地址：[aliyun_java_sdk_sts_20150825.zip](#)

更新日志:

- 发布AssumeRole接口

PHP SDK

示例

```
<?php
/*
 * 在您使用STS SDK前，请仔细阅读RAM使用指南中的角色管理部分，并阅读STS API文档
 * RAM/STS 文档：https://help.aliyun.com/product/8315075\_ram.html
 *
 */
```

```

include_once 'aliyun-php-sdk-core/Config.php';
use Sts\Request\V20150401 as Sts;

// 你需要操作的资源所在的region，STS服务目前只有华东 1 (杭州) 节点可以签发Token，签发出的Token在所有Region都可用
// 只允许子用户使用角色
$iClientProfile = DefaultProfile::getProfile("cn-hangzhou", "<access-key-id>", "<access-key-secret>");
$client = new DefaultAcsClient($iClientProfile);

// 角色资源描述符，在RAM的控制台的资源详情页上可以获取
$roleArn = "<role-arn>";

// 在扮演角色(AssumeRole)时，可以附加一个授权策略，进一步限制角色的权限；
// 详情请参考《RAM使用指南》
// 此授权策略表示读取所有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
// 您可以使用您的客户的ID作为会话名称
$request->setRoleSessionName("client_name");
$request->setRoleArn($roleArn);
$request->setPolicy($policy);
$request->setDurationSeconds(3600);
$response = $client->doAction($request);
print_r("\r\n");
print_r($response);

?>
    
```

PHP SDK 开发包

PHP SDK开发包(2015-09-01) 版本号2.1.6

下载地址：[aliyun_sts_php_sdk_2.1.6.zip](#)

更新日志:

- 发布AssumeRole接口

.Net SDK

示例

```
using System;
using Aliyun.Acs.Core;
using Aliyun.Acs.Core.Profile;
using Aliyun.Acs.Sts.Model.V20150401;

namespace sts_net_sdk_sample
{
    class Program
    {
        static void Main(string[] args)
        {
            // 构建一个 Aliyun Client, 用于发起请求
            // 构建Aliyun Client时需要设置AccessKeyId和AccessKeySevcret
            // STS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AssumeRole请求
            AssumeRoleRequest request = new AssumeRoleRequest();
            // 指定角色Arn
            request.RoleArn = "<role-arn>";
            request.RoleSessionName = "<role-session-name>";

            // 可以设置Token有效期, 可选参数, 默认3600秒;
            // request.DurationSeconds = 3600;
            // 可以设置Token的附加Policy, 可以在获取Token时, 通过额外设置一个Policy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();
    }
}
}

```

.Net SDK 开发包

.Net SDK开发包(2015-11-07) 版本号2.1.6

Java SDK下载地址 : aliyun_sts_net_sdk_2.1.6.zip

更新日志:

- 发布AssumeRole接口

Python SDK

示例

```

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

from aliyunsdkcore import client
from aliyunsdksts.request.v20150401 import AssumeRoleRequest

# 构建一个 Aliyun Client, 用于发起请求
# 构建Aliyun Client时需要设置AccessKeyId和AccessKeySevcret
# STS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"AssumeRole"请求
request = AssumeRoleRequest.AssumeRoleRequest()
# 指定角色
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(request)

print response
```

Python SDK

通过pip安装

```
pip install aliyun-python-sdk-sts
```

下载

- aliyun-python-sdk-core: <http://mirrors.aliyun.com/pypi/packages/source/a/aliyun-python-sdk-core/aliyun-python-sdk-core-2.0.15.tar.gz>
- aliyun-python-sdk-sts: <http://mirrors.aliyun.com/pypi/packages/source/a/aliyun-python-sdk-sts/aliyun-python-sdk-sts-2.1.6.tar.gz>

Release Notes

Version: 2.1.6

2015年11月10日发布

- STS Python SDK 第一个版本, 覆盖STS所有API

Node.js SDK

示例

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

// 构建一个 Aliyun Client, 用于发起请求
```

```
// 构建Aliyun Client时需要设置AccessKeyId和AccessKeySevcret
// STS是Global Service, API入口位于华东 1 (杭州), 这里使用sts API的主地址
var sts = new ALY.STS({
    accessKeyId: "<access-key-id>",
    secretAccessKey: "<access-key-secret>",
    endpoint: 'https://sts.aliyuncs.com',
    apiVersion: '2015-04-01'
});

// 构造AssumeRole请求
sts.assumeRole({
    Action: 'AssumeRole',
    // 指定角色Arn
    RoleArn: '<role-arn>',
    //设置Token的附加Policy, 可以在获取Token时, 通过额外设置一个Policy进一步减小Token的权限;
    //Policy: '{"Version": "1", "Statement": [{"Effect": "Allow", "Action": "*", "Resource": "*}]}'
    //设置Token有效期, 可选参数, 默认3600秒;
    //DurationSeconds: 3600,
    RoleSessionName: 'RoleSessionName'
}, function (err, res) {
    console.log(err, res);
});
```

Node.js SDK

通过npm安装

```
npm install aliyun-sdk
```

Release Notes

Version: 1.6.2

2015年11月10日发布

- STS Node.js SDK 第一个版本, 覆盖STS所有API