

Developer Guide

SDK Reference - STS

SDK Reference - STS

Java SDK

Preface

Introduction

This document describes how to install and use STS Java SDK. This document assumes that you have already subscribed to the Alibaba Cloud RAM service and created an Access Key ID and Access Key Secret for the primary account or the RAM account.

Before using STS Java SDK, read the [RAM User Guide](#) carefully to understand the meanings of the primary account, RAM account, federated account, and short-term access credentials, and STS application scenarios.

For details about STS API, refer to the [STS API Documentation](#).

Version revisions

Java SDK (2015-08-25) Version 2.1.6

- Release the AssumeRole interface

Installation

Use JAR package in Eclipse

Steps:

- Download STS Java SDK from Alibaba Cloud website
- Decompress the file
- Copy aliyun-java-sdk-core-`<versionId>`.jar and aliyun-java-sdk-sts-`<versionId>`.jar from the decompressed folder to your project folder
- In Eclipse, right-click and choose Project > Properties > Java Build Path > Add JARs
- Select all the JAR files you copied
- After completing the steps above, You can use STS Java SDK in the project

The STS Java SDK in other IDEs is configured in a similar way.

Use SDK in Maven projects

Add relevant aliyun-java-sdk dependency. Take version 2.1.6 as an example. Type the following content in the `<dependencies>` tab:

```
<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>
```

Getting started

Create and manage roles

To use STS' s AssumeRole interface, you need to create and manage roles in RAM. For details, refer to [RAM Role Management] (#)

Use Maven to create a project

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

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

Modify the generated pom.xml, and add relevant aliyun-java-sdk dependency. Take version 2.1.6 as an example. Type the following content in the “dependencies” tab:

```
<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 is added to <https://maven-repository.com/artifact/com.aliyun>

You need not to set settings.xml of the Maven.

Sample code used in aliyun-java-sdk-sts

- NOTE: Modify the value of accessKeyId and accessKeySecret to a valid value.
- In the com/aliyun/sts/sample/ directory, create the Java' s source code StsServiceSample.java and type the following content:

```
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 {
    // Currently, only the "cn-hangzhou" region is available. Do not fill in another region's value
    public static final String REGION_CN_HANGZHOU = "cn-hangzhou";
    // Current STS API version
    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 {
            // Construct an Aliyun Client for initiating an OpenAPI request
            IClientProfile profile = DefaultProfile.getProfile(REGION_CN_HANGZHOU, accessKeyId, accessKeySecret);
            DefaultAcsClient client = new DefaultAcsClient(profile);
```

```

// Create an AssumeRoleRequest and set the request parameters
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);

// Initiate a request and obtain the response
final AssumeRoleResponse response = client.getAcsResponse(request);

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

public static void main(String[] args) {
// Only an RAM account (sub-account) can call the AssumeRole interface
// The AccessKeys of an Alibaba Cloud primary account cannot be used to initiate an AssumeRole request
// You need to create an RAM account on RAM console, and create an AccessKeys for the account
String accessKeyId = "O*****F";
String accessKeySecret = "Y*****U";

// AssumeRole API request parameters include RoleArn, RoleSessionName, Policy, and DurationSeconds

// You need to obtain the RoleArn on RAM console
String roleArn = "acs:ram::145883****900618:role/ossadminrole";

// RoleSessionName is the session name of a temporary Token. You need to specify the parameter to identify your
// account. The parameter is mainly used for auditing purpose or distinguishing to which account the Token is issued
// However, attention should be paid to the length and rule of RoleSessionName. You cannot use space, but only '-'
//, '_', letters, and numbers
// For details rules, refer to the format requirements in API Documentation
String roleSessionName = "alice-001";

// How to customize your 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" +
"}";

// You must enter HTTPS here

```

```
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());
}

}
```

Compile and run sample code

Compile

```
mvn install
```

Run

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

Download SDK

Java SDK

- Alibaba Cloud SDK channel: <https://develop.aliyun.com/tools/sdk?#/java>
- GitHub address: <https://github.com/aliyun/aliyun-openapi-java-sdk/tree/master/aliyun-java-sdk-sts>

Update logs:

Java SDK (10/09/2017) Version 3.0.0

- Updated SDK Core to version 3.0.0.

Java SDK (08/25/2015) Version 2.1.6

- Released the AssumeRole API.

PHP SDK

Samples

```
<?php
/*
 * Before using STS SDK, read description on role management in the RAM Manual, and read the STS API
 * Documentation
 *
 */

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

// Currently, STS can only issue a Token for resources required for your operations in the East China 1 (Hangzhou)
// region, but the Token issued can be used in all regions
// Only sub-account can use the role
$iClientProfile = DefaultProfile::getProfile("cn-hangzhou", "<access-key-id>", "<access-key-secret>");
$client = new DefaultAcsClient($iClientProfile);

// You can obtain the role resource descriptor from the resource details page on RAM console
$roleArn = "<role-arn>";

// When assuming a role (AssumeRole), you can assign an authorization policy to further limit the role permission.
// For details, refer to the "RAM Use Guide"
// The authorization policy indicates the read-only permission on all OSS resources
$policy=<<<POLICY
{
  "Statement": [
  {
    "Action": [
      "oss:Get*",
      "oss:List*"
    ],
    "Effect": "Allow",
```

```
"Resource": "*"
}
],
"Version": "1"
}
POLICY;

$request = new Sts\AssumeRoleRequest();
// RoleSessionName indicates the session name of a temporary ID which is used to distinguish different temporary
temporary IDs
// You can use your customer's ID as the session name
$request->setRoleSessionName("client_name");
$request->setRoleArn($roleArn);
$request->setPolicy($policy);
$request->setDurationSeconds(3600);
$response = $client->doAction($request);
print_r("\r\n");
print_r($response);

?>
```

Download SDK

PHP SDK

PHP SDK (2015-09-01) Version 2.1.6

Go to the website [aliyun_sts_php_sdk_2.1.6.zip](#) to download the development kit

Updates:

- Release the AssumeRole interface

.Net SDK

Samples

```
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)
        {
            // Construct an Aliyun Client for initiating a request
            // Set AccessKeyId and AccessKeySecret when constructing the Aliyun Client
            // STS is a Global Service, and its API ingress is located in the East China 1 (Hangzhou) region. Enter "cn-hangzhou"
            // in "Region"
            IClientProfile profile = DefaultProfile.GetProfile("cn-hangzhou", "<access-key-id>", "<access-key-secret>");
            DefaultAcsClient client = new DefaultAcsClient(profile);

            // Construct an "AssumeRole" request
            AssumeRoleRequest request = new AssumeRoleRequest();
            // Specify the role Arn
            request.RoleArn = "<role-arn>";
            request.RoleSessionName = "<role-session-name>";

            // Set the Token's period of validity. It is an optional parameter with a default value of 3,600s.
            // request.DurationSeconds = 3600;
            // Set an additional policy for the Token so as to further limit the Token permission when obtaining the token.
            // 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 expiration time, and time for the server to return UTC, which is displayed in Beijing Time.
                Console.WriteLine("Expiration: " + DateTime.Parse(response.Credentials.Expiration).ToLocalTime());
            }
            catch (Exception ex)
            {
                Console.WriteLine(ex.ToString());
            }
            Console.ReadLine();
        }
    }
}
```

Download SDK

.Net SDK

.Net SDK (2015-11-07) Version 2.1.6 (.net 4.5)

Go to the website aliyun_sts_net_sdk_2.1.6.zip to download the Java SDK

Source Code: <https://github.com/aliyun/aliyun-openapi-net-sdk.git>

Updates:

- Release the AssumeRole interface

Python SDK

Sample

```
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)
        {
            // Construct an Aliyun Client for initiating a request
            // Set AccessKeyID and AccessKeySevcret when constructing the Aliyun Client
            // STS is a Global Service, and its API ingress is located in the East China 1 (Hangzhou) region. Enter "cn-hangzhou"
            in "Region"
```

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

// Construct an "AssumeRole" request
AssumeRoleRequest request = new AssumeRoleRequest();
// Specify the role Arn
request.RoleArn = "<role-arn>";
request.RoleSessionName = "<role-session-name>";

// Set the Token's period of validity. It is an optional parameter with a default value of 3,600s.
// request.DurationSeconds = 3600;
// Set an additional policy for the Token so as to further limit the Token permission when obtaining the token.
// 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 expiration time, and time for the server to return UTC, which is displayed in Beijing Time.
    Console.WriteLine("Expiration: " + DateTime.Parse(response.Credentials.Expiration).ToLocalTime());
}
catch (Exception ex)
{
    Console.WriteLine(ex.ToString());
}
Console.ReadLine();
}
```

Installation

.Net SDK

.Net SDK (2015-11-07) Version 2.1.6 (.net 4.5)

Go to the website aliyun_sts_net_sdk_2.1.6.zip to download the Java SDK

SourceCode:<https://github.com/aliyun/aliyun-openapi-net-sdk.git>

Updates:

- Release the AssumeRole interface

Node.js SDK

Samples

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

// Construct an Aliyun Client for initiating a request
// Set AccessKeyID and AccessKeySevcret when constructing the Aliyun Client
// STS is a Global Service, and its API ingress is located in the East China 1 (Hangzhou) region. The master IP address
// of STS API is used
var sts = new ALY.STS({
  accessKeyId: "<access-key-id>",
  secretAccessKey: "<access-key-secret>",
  endpoint: 'https://sts.aliyuncs.com',
  apiVersion: '2015-04-01'
});

// Construct an "AssumeRole" request
sts.assumeRole({
  Action: 'AssumeRole',
  // Specify the role Arn
  RoleArn: '<role-arn>',
  //Set an additional policy for the Token so as to further limit the Token permission when obtaining the token.
  //Policy: '{"Version":"1","Statement":[{"Effect":"Allow", "Action": "*", "Resource": "*"}]}',
  //Set the Token's period of validity. It is an optional parameter with a default value of 3,600s;
  //DurationSeconds: 3600,
  RoleSessionName: 'RoleSessionName'
}, function (err, res) {
  console.log(err, res);
});
```

Installation

NPM-based installation

```
npm install aliyun-sdk
```

Release Notes

Version: 1.6.2

Released on November 10, 2015

- The first version of STS Node.js SDK contains all APIs of STS