Alibaba Cloud
Resource Access Management

API 参考

Document Version20190220

目次

1 概要

1.1 用語集

用語	説明	補足
Account	アカウント	Alibaba Cloudアカウント
User	ユーザー	アカウントに属するRAMユーザー
Group	グループまたはユーザーグループ	アカウントに属するRAMユーザーグループ
Policy	権限付与ポリシー	権限付与ポリシーを説明する時に使用される言語
AK (Access Key)	アクセスキー	AccessKeyID と AccessKeySecret で構成されるアクセスキー。Alibaba CloudサービスAPIリクエストでのID認証に使用されます。
MFA (Multi-Factor Authentication)	多要素認証	アカウントまたはRAMユーザーにMFAデバイスをバインドすることで、セキュリティを強化します。

2 呼び出し方式

2.1 リクエストの構造

サービスアドレス

RAM サービスの API アクセス URL は次のとおりです：

```
https://ram.aliyuncs.com
```

通信プロトコル

通信のセキュリティを確保するために、RAM サービスは、安全な HTTPS チャネルでのみリクエストを送信します。

HTTP リクエスト方式

HTTP GET/POST リクエストを送信できます。 この方式では、リクエスト URL にリクエストパラメータを含める必要があります。

リクエストパラメータ

リクエストごとに Action パラメータを使用して、実行する操作を指定する必要があります。また、そのリクエストには、パブリックパラメータとインターフェイスサービスパラメータを追加する必要もあります。

文字エンコーディング

リクエストと返された結果は、UTF-8 文字セットでエンコードされます。

2.2 パブリックパラメータ

Format

- 名称：Format
- データ型：String
- 必須項目：いいえ
- 説明：戻り値の型。「JSON」と「XML」の2種類のタイプに対応しています。デフォルトでは「XML」です。

Version

- 名称：Version

- データ型：String
- 必須項目：はい
- 説明：APIバージョン番号。 形式：「YYYY-MM-DD」。現在の最新バージョンは「2015-05-01」です。

AccessKeyId

- 名称：AccessKeyId
- データ型：String
- 必須項目：はい
- 説明：アクセスキーID。

Signature

- 名称：Signature
- データ型：String
- 必須項目：はい
- 説明：メッセージの署名。

SignatureMethod

- 名称：SignatureMethod
- データ型：String
- 必須項目：はい
- 説明：署名方式。 現在は、「HMAC-SHA1」のみに対応しています。

SignatureVersion

- 名称：SignatureVersion
- データ型：String
- 必須項目：はい
- 説明：署名アルゴリズムのバージョン。 現在のバージョンは「1.0」です。

SignatureNonce

- 名称：SignatureNonce
- データ型：String
- 必須項目：はい
- 説明：一意の乱数で、ネットワーク反射攻撃を防ぎます。 リクエストごとに異なる乱数を使用する必要があります。

Timestamp

- 名称：Timestamp
- データ型：String
- 必須項目：はい
- 説明：クエストのタイムスタンプ。 日付形式はISO8601に従っており、UTC時刻を使用しています。 タイムスタンプ形式の例は次の通りです：

```
YYYY-MM-DDThh:mm:ssZ
```

例：2013-01-10T12:00:00Z (北京現地時間 1/10/2013 20:00:00と同じ)

リクエストの例

```
https://ram.aliyuncs.com/
          Format=xml
          &Version=2015-05-01
          &Signature=Pc5WB8gokVn0xfeu%2FZV%2BiNM1dgI%3D
          &SignatureMethod=HMAC-SHA1
          &SignatureNonce=15215528852396
          &SignatureVersion=1.0
          &AccessKeyId=key-test
          &Timestamp=2012-06-01T12:00:00Z
          …
```

パブリックリターンパラメータ

インターフェイスを呼び出すリクエストを送信するたびに、成功したかどうかに関係なく、一意の識別コード 「RequestId」) が返されます。 このパラメータは、各リクエストの特定に使用されます。

例

- XML形式

```
<? xml version="1.0" encoding="utf-8"? >
          <!—Result root node-->
          <Interface name+response>
          <!—Return request tag-->
          <RequestId>4C467B38-3910-447D-87BC-AC049166F216</RequestId
>
          <!—Return result data-->
          </Interface name+response>
```

- JSON形式

```
          "RequestId": "4C467B38-3910-447D-87BC-AC049166F216"
          /* Return result data */
```

2.3 返された結果の処理

API サービスを呼び出した後、データは一様な形式で返されます。 返された HTTP ステータスコードが 2xx という形式の場合、呼び出しは成功です。 返された HTTP ステータスコードが 4xx または 5xx の場合、呼び出しは失敗です。 呼び出しが成功した場合、データは主に XML と JSON の 2 つの形式で返されます。 外部システムはリクエストの送信時に、パラメータで返されるデータの形式を指定できます。デフォルトではXML形式です。 本ドキュメントでは、返される結果の例を見やすい形式で表示しています。 実際の結果は、改行やインデントなどで体裁が整えられているわけではありません。

呼び出し成功の例

- XML形式

  XML 形式で返された結果には、リクエストが成功したかどうかを示すメッセージと、特定のサービスデータが含まれます。 例：

```
<? xml version="1.0" encoding="utf-8"? >
<!—Result root node-->
<Interface name+response>
    <!—Return request tag-->
    <RequestId>4C467B38-3910-447D-87BC-AC049166F216</RequestId>
    <!—Return result data-->
</Interface name+response>
```

- JSON形式

```
    "RequestId": "4C467B38-3910-447D-87BC-AC049166F216",
    /* Return result data */
```

呼び出し失敗の例

インターフェイス呼び出しでエラーが発生した場合、結果が返されません。 エラーの原因は、添付の付録の*#unique_7*で確認できます。

呼び出しでエラーが発生した場合、HTTP リクエストに対して 4xx または 5xx という形式の HTTP ステータスコードが返されます。 返されるメッセージには、特定のエラーコードとエラーメッセージが含まれます。 また、メッセージには、全体で一意な「RequestId」と、リクエストした「HostId」も含まれています。 エラーの原因を特定できない場合は、Alibaba Cloud カスタマーサービスにご連絡ください。その際、よりスムーズに問題を解決できるよう「HostId」と「RequestId」をご提示ください。

- XML形式

```
<? xml version="1.0" encoding="UTF-8"? >
<Error>
   <RequestId>8906582E-6722-409A-A6C4-0E7863B733A5</RequestId>
   <HostId>ram.aliyuncs.com</HostId>
   <Code>InvalidParameter</Code>
   <Message>The specified parameter "Action or Version" is not valid
.</Message>
</Error>
```

- JSON形式

```
    "RequestId": "7463B73D-35CC-4D19-A010-6B8D65D242EF",
    "HostId": "ram.aliyuncs.com",
    "Code": "InvalidParameter",
    "Message": "The specified parameter \"Action or Version\" is not
 valid."
```

2.4 署名のしくみ

STSサービスは、アクセスリクエストするたびにID認証を行います。 それにより、HTTPまたはHTTPSを通じて送信する場合は、リクエストには署名情報が含まれる必要があります。 RAMはAccessKey IDとAccess Key Secretを通じての対称暗号化という方式でリクエスト送信者の個人情報を認証します。 AccessKey IDとAccesKeySecretはAlibabaクラウドよりリクエスト送信者に対して公式に発行されます（クエスト送信者はAlibabaクラウドの公式Webサイトでこれらの情報を申請し、管理できます）。 AccessKey IDはリクエスト送信者のIDを示します。 AccessKey Secretは署名文字列を暗号化するに使用される秘密鍵であり、サーバーで署名文字列を認証する際にも用いられます。 この情報は機密として厳密に取り扱い、Alibabaクラウドと認証済みリクエスト送信者以外には知られないようにしてください。

リクエスト署名プロセス

1. リクエストパラメータを使用して、正規化クエリ文字列を作成します。

   a. すべてのリクエストパラメータ (「パブリックリクエストパラメーター」と、このドキュメントで取り上げる任意のリクエストインターフェイスに対するユーザー定義パラメータが含まれますが、「パブリックリクエストパラメータ」に説明のあるSignatureパラメータは含まれません) をパラメーター名のアルファベット順に並べ替えます。

   > 注：
   > 「GET」メソッドを使用してリクエストを送信する際に、これらのパラメータはリクエスト URL (URLのアンパーサンド 「&」に続く疑問符 「?」の後ろの部分) に含まれます。

b. 各リクエストパラメータの名前と値をエンコードします。 UTF-8 文字セットを使用したURL エンコードが必要です。 URL エンコーディングのルールは次のとおりです。

- 大文字 (A ～ Z)、小文字 (a ～ z)、整数 (0 ～ 9)、および一部の記号 (ハイフン “- “、アンダーバー “_” 、ピリオド “.” 、チルダ “~” など) はエンコードされません。
- 他の文字は “%XY” 形式でエンコードします。この XY は、文字の ASCII コードを16 進表記することを意味します。 たとえば、二重引用符 (“) は “%22” です。
- 半角スペース ( ) はプラス記号 “+” ではなく、”%20” としてエンコードされることに注意してください。

> 注 :
> 一般的に、URL エンコーディングされたライブラリ(Javaの「java.net.URLEncoder」など) は、「application/x-www-form-urlencoded」形式のMIMEタイプのルールに基づいてエンコードされます。 このエンコーディング方法を使用するには、前述のエンコーディングルールに合わせて、エンコードされた文字列内のプラス記号 「+」を「%20」に、アスタリスク「*」を「%2A」に直接置き換え、「%7E」をチルダ「~」に戻します。

c. エンコードしたパラメーター名と値を半角の等号「=」で連結します。

d. 等号で結んだパラメーター名と値のペアを、アンパーサンド「&」を使用してパラメーター名のアルファベット順に連結して、正規化クエリ文字列を作成します。

2. 次のルールに従って正規化クエリ文字列を使用し、署名計算用の文字列を作成します：

```
StringToSign= HTTPMethod + “&” + percentEncode(“/”) + ”&” +
percentEncode(CanonicalizedQueryString)
```

ここで「HTTPMethod」は、リクエストの送信に使用する HTTP メソッド (“GET” など) です。 「percentEncode (「/」)」 は、1.b で説明したURLエンコーディングルールに従って文字「/」をエンコードした値 (「%2F」) です。

「percentEncode(CanonicalizedQueryString)」 は、1.b のURLエンコーディングルールに従ってエンコードした正規化クエリ文字列です。

3. *RFC2104*の定義に基づいて、署名計算のための文字列を使用して署名のHMAC値を計算します。 注意: 署名計算に使用するキーは、 Access Key Secretにアンパーサンド “&” (ASCII: 38) を付加したもので、ハッシュアルゴリズムSHA1に基づいています。
4. Base64コーディングルールに基づいて HMAC 値を文字列にエンコードし、署名値を取得します。
5. 取得した署名値を「Signature」パラメータとしてリクエストパラメータに追加し、リクエスト署名プロセスを完了します。 注意: 取得した署名値は、最終的なリクエストパラメーター値

として STS サーバーに送信する前に、他のパラメーターと同様、RFC3986ルールに基づいてURLエンコーディングする必要があります。

例

「CreateUser」を例として使用した場合、署名前のリクエスト URL は次のようになります：

```
https://ram.aliyuncs.com/?UserName=test&SignatureVersion=1.0&Format=
JSON&Timestamp=2015-08-18T03%3A15%3A45Z&AccessKeyId=testid&SignatureM
ethod=HMAC-SHA1&Version=2015-05-01&Action=CreateUser&SignatureNonce=
6a6e0ca6-4557-11e5-86a2-b8e8563dc8d2
```

対応する「StringToSign」は:

```
GET&%2F&AccessKeyId%3Dtestid%26Action%3DCreateUser%26Format%3DJSON%
26SignatureMethod%3DHMAC-SHA1%26SignatureNonce%3D6a6e0ca6-4557-11e5-
86a2-b8e8563dc8d2%26SignatureVersion%3D1.0%26Timestamp%3D2015-08-18T03
%253A15%253A45Z%26UserName%3Dtest%26Version%3D2015-05-01
```

AccessKeyIdパラメータ値を「testid」、 AccessKeySecretパラメータ値を「testsecret」とすると、HMACの計算に使用するKeyは「Testsecret&」、計算結果の署名値は次のようになります：

```
kRA2cnpJVacIhDMzXnoNZG9tDCI%3D
```

署名付きリクエストURLは次のようになります(「Signature」 パラメータが追加されています)。

```
https://ram.aliyuncs.com/?UserName=test&SignatureVersion=1.0&Format=
JSON&Timestamp=2015-08-18T03%3A15%3A45Z&AccessKeyId=testid&SignatureM
ethod=HMAC-SHA1&Version=2015-05-01&Signature=kRA2cnpJVacIhDMzXnoN
ZG9tDCI%3D&Action=CreateUser&SignatureNonce=6a6e0ca6-4557-11e5-86a2-
b8e8563dc8d2
```