

Cloud Object Storage

トラブルシューティング

製品ドキュメント



Tencent Cloud

Copyright Notice

©2013-2023 Tencent Cloud. All rights reserved.

Copyright in this document is exclusively owned by Tencent Cloud. You must not reproduce, modify, copy or distribute in any way, in whole or in part, the contents of this document without Tencent Cloud's the prior written consent.

Trademark Notice

 Tencent Cloud

All trademarks associated with Tencent Cloud and its services are owned by Tencent Cloud Computing (Beijing) Company Limited and its affiliated companies. Trademarks of third parties referred to in this document are owned by their respective proprietors.

Service Statement

This document is intended to provide users with general information about Tencent Cloud's products and services only and does not form part of Tencent Cloud's terms and conditions. Tencent Cloud's products or services are subject to change. Specific products and services and the standards applicable to them are exclusively provided for in Tencent Cloud's applicable terms and conditions.

カタログ：

トラブルシューティング

RequestId取得の操作ガイド

パブリックネットワーク経由でのCOSへのファイルアップロード速度の遅さ

COSへのアクセス時に403エラーコードが返される

リソースアクセス異常

ファイルへのアクセス時に404 NoSuchKeyが返される

ビデオへのアクセス時に正常に再生できない

同一リンクからアクセスしたファイルの異常

CDNドメイン名を使用したCOSへのアクセス時にHTTP ERROR 403が返される

HTTPプロトコルを使用してカスタムオリジンサーバードメイン名にアクセスした際にエラーが発生する

POST Objectの一般的な異常

トラブルシューティング

RequestId取得の操作ガイド

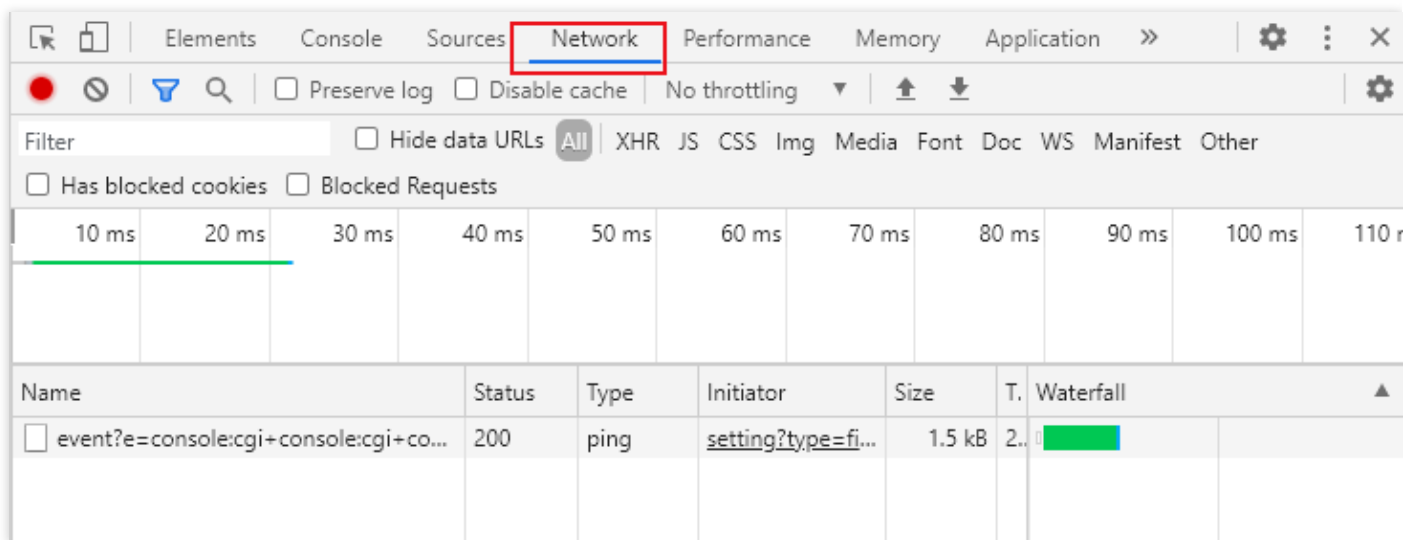
最終更新日：：2023-05-23 15:46:59

概要

Cloud Object Storage（COS）サービスにおいて、毎回のリクエスト送信時に、COSサーバーはリクエストに対し1つのID、すなわちRequestIdを生成します。ここでは、様々なケースにおけるRequestIdの取得方法についてご説明します。

コンソール上でブラウザを通じて取得

1. [COSコンソール](#)にログインし、左側ナビゲーションバーの**バケットリスト**をクリックし、バケットリストページに進みます。
2. アクセスしたいバケットをクリックして進みます。
3. **F12** キーを押し、ブラウザの開発者ツールページに進みます。
4. 開発者ツールの上方の**Network**をクリックします。



5. ダウンロードしたいファイル名の右側の**ダウンロード**をクリックし、開発者ツールページでダウンロードしたいファイル名を入力してフィルタリングを行います。ファイルを選択し、**Headers**をクリックし、**Response Headers**エリアでRequestId情報を取得します。

ファイルアクセス失敗時の取得

ファイルへのアクセスが失敗した際、ページに表示される、返されたXML情報の中から**RequestId**ノード情報を取得できます。

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```

▼<Error>
  <SCRIPT id="allow-copy_script">(function agent() { let unlock = false document.addEventListener('allow_copy',
  'contextmenu', 'selectstart', 'mousedown', 'mouseup', 'mousemove', 'keydown', 'keypress', 'keyup', ] const rej
  (e.stopImmediatePropagation) e.stopImmediatePropagation() } } copyEvents.forEach((evt) => { document.documentE
  <Code>AccessDenied</Code>
  <Message>Access Denied.</Message>
  <Resource>00000000-1251668577.cos.ap-chengdu.myqcloud.com/1920x1200.jpg</Resource>
  <RequestId>NjBjYWNmYmFmJjJiMjU4NjRfNGJhN180YT< </RequestId>
  <TraceId>OGVmYzZlMmQzYjAzOWNhODk0NTkRMTB1OWVmMDAxODc0UWRkZjJkOzDM1NmI1M2E2MTRlY2MzZDhmNmI5MWI1OWI2NmQ0YjJkZWE3N
  </Error>
  
```

次の操作を行って取得することもできます。

1. F12 キーを押し、ブラウザの開発者ツールページに進みます。
2. ページ上方の**Network**をクリックし、Allタイプを選択すると、Response Headersの中からRequestIdフィールド情報を見つけることができます。

The screenshot shows the browser's developer tools. On the left, the XML response is displayed, with the `<RequestId>` node highlighted in red. On the right, the 'Network' tab is active, showing the 'Headers' for the failed request. The 'x-cos-request-id' header is highlighted in red, showing the same value as the RequestId in the XML: `NjBjYWNmYmFmJjJiMjU4NjRfNGJhN180YT`.

SDKによる取得

SDKに含まれるインターフェースは非常に多く、すべてのインターフェースの例を挙げることはできないため、すべてのSDKについてファイルのアップロードを例にとり、現在の操作のRequestIdの取得方法を示します。

.NET SDKによる取得

```
try
{
    string bucket = "examplebucket-1250000000"; //バケット、形式: BucketName-APPID
    string cosPath = "test.cs"; // オブジェクトキー
    byte[] data = System.Text.Encoding.Default.GetBytes("Hello COS"); // バイナリーデータ
    PutObjectRequest putObjectRequest = new PutObjectRequest(bucket, cosPath, data);

    PutObjectResult result = cosXml.PutObject(putObjectRequest);
    string requestId = result.responseHeaders.GetValueOrDefault("x-cos-request-id")[0];
    Console.WriteLine(requestId);
}
catch (COSXML.CosException.CosClientException clientEx)
{
    //リクエストに失敗しました
    Console.WriteLine("CosClientException: " + clientEx);
}
catch (COSXML.CosException.CosServerException serverEx)
{
    //リクエストに失敗しました
    Console.WriteLine("CosServerException: " + serverEx.GetInfo());
}
```

Go SDKによる取得

```
package main

import (
    "context"
    "fmt"
    "net/http"
    "net/url"
    "strings"
    "github.com/tencentyun/cos-go-sdk-v5"
)

func main() {
    // examplebucket-1250000000およびCOS_REGIONを真の情報に変更します
    u, _ := url.Parse("https://examplebucket-1250000000.cos.COS_REGION.myqcloud.com")
    b := &cos.BaseURL{BucketURL: u}
    c := cos.NewClient(b, &http.Client{
        Transport: &cos.AuthorizationTransport{
            SecretID: "SECRETID",
            SecretKey: "SECRETKEY",
        },
    })
}
```

```
},
})
// オブジェクトキー (Key) はオブジェクトのバケット内での固有識別子です。
// 例えば、オブジェクトのアクセスドメイン名`examplebucket-1250000000.cos.COS_REGION.myq
cloud.com/test.go`の中で、オブジェクトキーはtest.goとなります
name := "test.go"
// 1.文字列によってオブジェクトをアップロードします
f := strings.NewReader("Hello COS")

_, err := c.Object.Put(context.Background(), name, f, nil)
if err != nil{
// error情報にはRequestIdフィールドが直接含まれます
panic(err)
}
requestId := response.Header.Get("X-Cos-Request-Id")
fmt.Println(requestId)
}
```

Java SDKによる取得

```
// 1 ユーザーID情報 (secretId、secretKey) を初期化します。
String secretId = "SECRETID";
String secretKey = "SECRETKEY";
COSCredentials cred = new BasicCOSCredentials(secretId, secretKey);
// 2 bucketのリージョンを設定します。COSリージョンの略称についてはhttps://cloud.tencent.c
om/document/product/436/6224をご参照ください
// clientConfigにはregion、https(デフォルトではhttp)、タイムアウト、プロキシなどを設定する
setメソッドが含まれます。使用にあたってはソースコードまたはよくあるご質問のJava SDKのパートを参
照できます
Region region = new Region("COS_REGION");
ClientConfig clientConfig = new ClientConfig(region);
// ここではhttpsプロトコルの設定と使用を推奨します
clientConfig.setHttpProtocol(HttpProtocol.https);
// 3 cosクライアントを生成します。
COSClient cosClient = new COSClient(cred, clientConfig);
// Bucketの命名形式はBucketName-APPIDです。ここに入力するバケット名は、この形式でなければな
りません
String bucketName = "examplebucket-1250000000";

String content = "Hello COS";
String key = "test.java";
PutObjectResult putObjectResult = cosClient.putObject(bucketName, key, content);
String requestId = putObjectResult.getRequestId();
System.out.println(requestId);
```

Python SDKによる取得

```
# -*- coding=utf-8
from qcloud_cos import CosConfig
from qcloud_cos import CosS3Client
import sys
import os
import logging

# 正常な状況のログレベルはINFOを使用し、特定が必要な場合はDEBUGに変更します。このときSDKは、サーバーとの通信情報をプリントアウトします
logging.basicConfig(level=logging.INFO, stream=sys.stdout)

# 1. secret_id, secret_key, regionなどを含む、ユーザーの属性を設定します。AppidはすでにCosConfigから削除されていますので、パラメータBucket中にAppidを含めてください。Bucketは、BucketName-Appidによって構成されます
secret_id = os.environ['COS_SECRET_ID'] # ユーザーのSecretIdです。サブアカウントのキーを使用し、権限承認は最小権限ガイドに従って行い、使用上のリスクを低減させることをお勧めします。サブアカウントキーの取得については、https://cloud.tencent.com/document/product/598/37140をご参照ください
secret_key = os.environ['COS_SECRET_KEY'] # ユーザーのSecretKeyです。サブアカウントのキーを使用し、権限承認は最小権限ガイドに従って行い、使用上のリスクを低減させることをお勧めします。サブアカウントキーの取得については、https://cloud.tencent.com/document/product/598/37140をご参照ください
region = 'ap-beijing' # ユーザーのregionに置き換えます。すでに作成済みのバケットが所属するregionについては、コンソールでhttps://console.tencentcloud.com/cos5/bucketをご参照ください
# COSがサポートするすべてのregionリストについては https://cloud.tencent.com/document/product/436/6224をご参照ください
token = None # パーマネントキーを使用する場合、tokenへの入力は不要です。一時キーを使用する場合は入力が必要です。一時キーの発行および使用ガイドについてはhttps://cloud.tencent.com/document/product/436/14048をご参照ください
scheme = 'https' # http/httpsプロトコルを指定使用してCOSにアクセスします。デフォルトはhttpsです。未入力でもかまいません

config = CosConfig(Region=region, SecretId=secret_id, SecretKey=secret_key, Token=token, Scheme=scheme)
client = CosS3Client(config)

try:
    response = client.put_object(
        Bucket='examplebucket-1250000000',
        Key='exampleobject',
        Body=b'abcdefg'
    )

# リクエストの正常なリターンについては、responseによりrequest-idを確認します
```



```
if 'x-cos-request-id' in response:
    print(response['x-cos-request-id'])

# リクエスト失敗については、異常によりrequest-idを確認します
except CosServiceError as e:
    print(e.get_request_id())
```

JavaScript SDKによる取得

```
cos.putObject({
    Bucket: 'examplebucket-1250000000', /* 必須 */
    Region: 'COS_REGION', /* 必須 */
    Key: 'test.js', /* 必須 */
    StorageClass: 'STANDARD',
    Body: 'Hello COS',
    onProgress: function(progressData) {
        console.log(JSON.stringify(progressData));
    }
}, function(err, data) {
    var requestId = (err || data).headers['x-cos-request-id'];
    console.log(requestId);
});
```

Node.js SDKによる取得

```
var COS = require('cos-nodejs-sdk-v5');
var cos = new COS({
    SecretId: 'SECRETID',
    SecretKey: 'SECRETKEY'
});

cos.putObject({
    Bucket: 'examplebucket-1250000000', /* 必須 */
    Region: 'COS_REGION', /* 必須 */
    Key: 'test.nodejs', /* 必須 */
    StorageClass: 'STANDARD',
    Body: Buffer.from('Hello COS'),
    onProgress: function(progressData) {
        console.log(JSON.stringify(progressData));
    }
}, function(err, data) {
    var requestId = (err || data).headers['x-cos-request-id'];
    console.log(requestId);
});
```

WeixinアプレットSDKによる取得

```
var COS = require('cos-wx-sdk-v5');
var cos = new COS({
  SecretId: 'SECRETID',
  SecretKey: 'SECRETKEY'
});

cos.putObject({
  Bucket: 'examplebucket-1250000000', /* 必須 */
  Region: 'COS_REGION', /* 必須 */
  Key: 'test.js', /* 必須 */
  StorageClass: 'STANDARD',
  Body: 'Hello COS',
  onProgress: function(progressData) {
    console.log(JSON.stringify(progressData));
  },
  function(err, data) {
    var requestId = (err || data).headers['x-cos-request-id'];
    console.log(requestId);
  }
});
```

PHP SDKによる取得

```
$secretId = "SECRETID"; //Tencent Cloud APIキー SecretId";
$secretKey = "SECRETKEY"; //Tencent Cloud APIキー SecretKey";
$region = "COS_REGION"; //デフォルトのバケットリージョンを設定
$cosClient = new Qcloud\Cos\Client(
  array(
    'region' => $region,
    'schema' => 'https', //プロトコルヘッダー。デフォルトではhttp
    'credentials'=> array(
      'secretId' => $secretId ,
      'secretKey' => $secretKey));
  # ファイルのアップロード
  ## putObject (アップロードインターフェース。最大5Gまでのファイルのアップロードをサポート)
  ### アップロードメモリ内の文字列
  try {
    $bucket = "examplebucket-1250000000"; //バケット名 形式: BucketName-APPID
    $key = "test.php"; //ここでのkeyはオブジェクトキーです。オブジェクトキーはオブジェクトのバケ
    ット内での固有識別子です
    $result = $cosClient->putObject(array(
      'Bucket' => $bucket,
      'Key' => $key,
      'Body' => 'Hello COS'));
    $requestId = $result['RequestId'];
```

```
print_r($requestId);
} catch (\Exception $e) {
echo "$e\n";
}
```

iOS SDKによる取得

```
QCloudCOSXMLUploadObjectRequest* put = [QCloudCOSXMLUploadObjectRequest new];
/** ローカルファイルパスは、URLを必ずfile:///で始めてください。形式は次のとおりです。
1. [NSURL URLWithString:@"file:///var/mobile/Containers/Data/Application/DBPF7490-D5U8-4ABF-A0AF-CC49D6A60AEB/Documents/exampleobject"]
2. [NSURL fileURLWithPath:@"var/mobile/Containers/Data/Application/DBPF7490-D5U8-4ABF-A0AF-CC49D6A60AEB/Documents/exampleobject"]
*/
NSURL* url = [NSURL fileURLWithPath:@"ファイルのURL"];
// バケット名はBucketName-Appidで構成され、COSコンソールで確認できます。 https://console.tencentcloud.com/cos5/bucket
put.bucket = @"examplebucket-1250000000";
// オブジェクトキーは、オブジェクトのCOS上での完全パスです。ディレクトリがある場合は、形式は「video/xxx/movie.mp4」です
put.object = @"exampleobject";
// アップロードする必要があるオブジェクトの内容です。NSData*またはNSURL*タイプの変数を渡すことができます
put.body = url;
// アップロードの進捗を監視します
[put setSendProcessBlock:^(int64_t bytesSent,
int64_t totalBytesSent,
int64_t totalBytesExpectedToSend) {
// bytesSent 今回送信する必要があるバイト数（大容量ファイルは数回に分けての送信が必要な可能性があります）
// totalBytesSent 送信済みバイト数
// totalBytesExpectedToSend 今回のアップロードで送信する必要がある総バイト数（すなわち1ファイルのサイズ）
}];
// アップロードの結果を監視します
[put setFinishBlock:^(QCloudUploadObjectResult *result, NSError *error) {
// requestidを取得します
[result.__originHTTPURLResponse__.allHeaderFields objectForKey:@"x-cos-request-id"];
}];
[put setInitMultipleUploadFinishBlock:^(QCloudInitiateMultipartUploadResult *
multipleUploadInitResult,
QCloudCOSXMLUploadObjectResumeData resumeData) {
// マルチパートアップロードの初期化が完了すると、このblockがコールバックされ、ここでresumeData、uploadidが取得できるようになります
NSString* uploadId = multipleUploadInitResult.uploadId;
```

```
});  
[[QCloudCOSTransferMangerService defaultCOSTransferManager] UploadObject:put];
```

Android SDKによる取得

```
// 1. 初期化TransferServiceです。同じ設定の場合、同一のTransferServiceを再利用しなければなりません  
TransferConfig transferConfig = new TransferConfig.Builder()  
.build();  
CosXmlServiceConfig cosXmlServiceConfig = new CosXmlServiceConfig.Builder()  
.setRegion(COS_REGION)  
.builder();  
CosXmlService cosXmlService = new CosXmlService(context, cosXmlServiceConfig, credentialProvider);  
TransferService transferService = new TransferService(cosXmlService, transferConfig);  
  
// 2. 初期化PutObjectRequest  
String bucket = "examplebucket-1250000000"; //バケット、形式: BucketName-APPID  
String cosPath = "exampleobject"; //オブジェクトのバケット内での位置識別子です。オブジェクトキーと呼ばれます  
String srcPath = "examplefilepath"; //ローカルファイルの絶対パス  
PutObjectRequest putObjectRequest = new PutObjectRequest(bucket,  
cosPath, srcPath);  
  
// 3. upload方法呼び出し、ファイルをアップロード  
final COSUploadTask uploadTask = transferService.upload(putObjectRequest);  
uploadTask.setCosXmlResultListener(new CosXmlResultListener() {  
@Override  
public void onSuccess(CosXmlRequest request, CosXmlResult result) {  
// アップロードに成功しました。ここでrequestIdを取得できます  
String requestId = result.getHeader("x-cos-request-id");  
}  
  
@Override  
public void onFail(CosXmlRequest request,  
CosXmlClientException clientException,  
CosXmlServiceException serviceException) {  
// CosXmlServiceExceptionさえあれば、異常時にのみrequestIdが得られます  
if (serviceException != null) {  
String requestId = serviceException.getRequestId();  
}  
}  
});
```

パブリックネットワーク経由でのCOSへの ファイルアップロード速度の遅さ

最終更新日：：2022-06-28 16:10:46

故障について

- **現象1**：
 - 会社のネットワークを使用してアップロードする場合は正常に転送されるが、ホームネットワークを使用してアップロードすると転送速度が遅くなる（8Mbps未満）。
 - 携帯電話の4Gネットワークを使用してアップロードする場合は正常に転送されるが、会社のネットワークを使用してアップロードすると転送速度が遅くなる（8Mbps未満）。
- **現象2**：カスタムドメイン名を使用してアップロードすると転送速度が遅くなる。

考えられる原因

- 現象1について：
 1. ネットワーク環境によってCOSアクセスの速度が異なる場合は、現在のネットワークキャリアやネットワーク環境に関係がある可能性があります。
 2. ネットワーク環境によってCOSアクセスの速度が異なる場合は、越境アクセスが原因の可能性があります。
- 現象2について：カスタムドメイン名CNAMEを他の製品に切り替え、再びCOSに戻します。例えばContent Delivery Network（CDN）、Cloud Virtual Machine（CVM）、セキュリティ製品などです。

ソリューション

- **現象1**が発生した場合は、クライアントのネットワーク環境をチェックすることによって自ら対応できます。操作の詳細については、[クライアントネットワークのトラブルシューティング](#)をご参照ください。
- **現象2**が発生した場合は、カスタムドメイン名の解決を変更する方法で転送中継リンクを減少させ、転送効率を向上させることができます。操作の詳細については、[カスタムドメイン名解決の変更](#)をご参照ください。

処理手順

クライアントネットワークのトラブルシューティング

1. 次のコマンドを実行し、IPアドレスのキャリアとクライアントネットワークキャリアが一致しているかどうかを確認します。

```
ping COSのアクセスドメイン名
```

例：

```
ping examplebucket-1250000000.cos.ap-beijing.mqcloud.com
```

- 出力する場合は、[手順3](#)を実行してください。
- NOの場合は、[手順2](#)を実行してください。

2. ブラウザがプロキシを設定しているかどうかをチェックします。Chromeブラウザを例にとります。



i. Chromeブラウザを開き、右上隅の **> 設定** をクリックし、設定ページを開きます。

ii. **アドバンス** をクリックし、「システム」バーで **お客様のコンピュータでのプロキシ設定** を選択し、OSの設定ウィンドウを開きます。

プロキシが設定されているかどうかをチェックします。

- YESの場合は、プロキシを無効にします。
 - NOの場合は、[手順3](#)を実行してください。
3. 使用しているWi-Fiルーターに速度制限が存在するかどうかをチェックします。
- YESの場合は、実際のニーズに応じて適宜リリースしてください。
 - NOの場合は、[手順4](#)を実行してください。

4. 現在のネットワークでのCOSへのアップロードの転送パフォーマンスをチェックします。

COSのCOSCMDツールを例に、20MBのオブジェクトのアップロードおよびダウンロードパフォーマンスをテストします。

```
coscmd probe -n 1 -s 20
```

次のような結果が返され、平均速度（Average）、最低速度（Min）、最高速度（Max）がそれぞれ得られます。

```
D:\work>coscmd probe -n 1 -s 20
Upload tmp_test_20M => cos://[redacted]/tmp_test_20M
[success]
Download cos://[redacted]/tmp_test_20M => tmp_test_20M
[success]
Success Rate: [1/1]
  20M TEST      Average      Min          Max
  Upload        12.6MB/s    12.6MB/s    12.6MB/s
  Download      10.2MB/s    10.2MB/s    10.2MB/s
```

5. ブラウザからネットワークスピードテストにアクセスし、手順4の結果も踏まえて、クライアントのネットワーク帯域幅占有率が上限に達していないかを確認します。

- 手順4の速度がクライアント帯域幅速度より低い場合は、お問い合わせください。
- 手順4の速度がクライアント帯域幅速度と同じであり、かつキャリアのコミットした帯域幅に達していない場合は、キャリアのカスタマーサービスに連絡してください。
- 手順4の速度がクライアント帯域幅速度と同じであり、かつキャリアのコミットした帯域幅に達している場合は、手順6を実行してください。

6. 国内のクライアントが海外ノードのbucketにアクセスしている、あるいは海外のクライアントが国内ノードのbucketにアクセスしている状況がないかを確認します。

- 「はい」の場合は、COSのグローバルアクセラレーション機能を使用することをお勧めします。
- 「いいえ」の場合は、お問い合わせください。

カスタムドメイン名解決の変更

1. カスタムドメイン名がCOSドメイン名に解決されているかどうかを確認します。

- 「はい」の場合は、お問い合わせください。
一般的なCOSドメイン名は次のとおりです。

```
XXX.cos.ap-beijing.myqcloud.com (COSデフォルトドメイン名)
XXX.cos.accelerate.myqcloud.com (COSグローバルアクセラレーションドメイン名)
XXX.cos-website.ap-beijing.myqcloud.com (COS静的ページドメイン名)
XXX.picbj.myqcloud.com (COS Cloud Infiniteデフォルトドメイン名)
```

- 「いいえ」の場合は、[手順2](#)を実行してください。
一般的な非COSドメイン名は次のとおりです。

```
XXX.file.myqcloud.comまたはXXX.cdn.dnsv1.com (Tencent Cloud CDNデフォルトドメイン名)  
XXX.w.kunlungr.com (aliyunCDNデフォルトドメイン名)
```

2. カスタムドメイン名のCNAMEを必要なCOSドメイン名に解決し、データのアップロードを行います。
例えば、`upload.mydomain.com cname XXX.cos.ap-beijing.myqcloud.com` などとします。具体的な操作については、[カスタムオリジンサーバードメイン名の有効化](#)をご参照ください。
3. クライアントのCOSドメイン名を変更します。
C#コードを例にとります。

```
CosXmlConfig config = new CosXmlConfig.Builder()  
.SetConnectionTimeoutMs(60000) //接続タイムアウト時間を設定します。単位はミリ秒、デフォルトでは45000msです  
.SetReadWriteTimeoutMs(40000) //読み取り/書き込みタイムアウト時間を設定します。単位はミリ秒、デフォルトでは45000msです  
.IsHttps(true) //デフォルトhttpsリクエストを設定します  
.SetAppid(appid) //Tencent Cloudアカウントのアカウント識別APPIDを設定します  
.SetRegion(region) //デフォルトのバケットリージョンを設定します  
.SetHost("XXXXXXX.com") //カスタムドメイン名を入力します  
.SetDebugLog(true) .Build(); //CosXmlConfigオブジェクトを作成します
```

その他のSDKの呼び出しについては、[SDKの概要](#)をご参照ください。

COSへのアクセス時に403エラーコードが返される

最終更新日：：2022-09-28 14:41:24

障害事象

- COS API、SDKでリソースをアップロードおよびダウンロードする際、403エラーコードが返されます。
- 一時キーまたはサブアカウントを使用してCOSリソースにアクセスする際、403エラーコードが返されます。
- COS bucketの設定を変更する際、403エラーコードが返されます。

障害分析の考え方

COSリクエストに対し403エラーコードが表示される場合は、以下のフローを参照してトラブルシューティングを行うことができます。

1. リクエストがCORSクロスドメインアクセスリクエストかどうかを確認します。失敗したクロスドメインリクエストには"[AccessForbidden](#)"が返されます。
2. リクエストがバケットのリンク不正アクセス防止設定にヒットしたかどうかを確認します。リンク不正アクセス防止ルールによって403エラーコードが発生した場合、COSは"[You are denied by bucket referer rule](#)"を返します。
3. リクエストが匿名リクエストかどうかを確認します。パブリック読み取りではないオブジェクトに対し、署名付きではないリクエストを送信した場合は"[Access Denied.](#)"が返されます。バケットまたはオブジェクトをパブリック読み取りに設定する必要がある場合は、[アクセス権限の設定](#)または[オブジェクトのアクセス権限の設定](#)をご参照ください。
4. リクエストキーおよびリクエスト署名が正しいかどうかを確認します。
 - i. 署名にSecretIdが正しく使用されていない場合、COSは"[InvalidAccessKeyId](#)"を返します。
 - ii. ローカル時刻が不正確、またはリクエスト時刻が署名の有効期限を超過していた場合、COSは"[RequestTimeTooSkewed](#)"または"[Request has expired](#)"を返します。
 - iii. 生成した署名の計算方法に問題が存在する場合、COSは"[SignatureDoesNotMatch](#)"を返します。
5. リクエストを送信したサブアカウントまたは一時キーに必要なアクセス権限が与えられているかどうかを確認します。
 - i. サブアカウントのアクセス権限を確認します。リソースへのアクセス権限が与えられていないサブアカウントからのリクエストに対しては、"[Access Denied.](#)"が返されます。
 - ii. 一時キーを使用して送信されたリクエストについては、一時キーの申請の際に入力したポリシーが一時キーのリクエストするリソースの範囲を制限している場合があります。詳細については、[一時キーの生成および](#)

[使用ガイド](#)をご参照ください。

6. リクエストしたオブジェクトがアーカイブタイプまたはディープアーカイブタイプかどうかを確認します。このタイプのリクエストに対し、COSは"**InvalidObjectState**"を返します。

障害の特定および処理

Messageが「Access Denied.」の場合

COSへのアクセスの際に次のメッセージが表示された場合、

```
<Code>AccessDenied</Code>
<Message>Access Denied.</Message>
```

次の操作を実行する必要があります。

1. [COSコンソール](#)にログインします。
2. 左側ナビゲーションバーで**バケットリスト**を選択し、バケット管理ページに進みます。
3. 操作したいバケットを見つけ、そのバケット名をクリックし、バケット設定ページに進みます。
4. 左側ナビゲーションバーで**権限管理**>**バケットアクセス許可**を選択し、バケットアクセス権限管理ページに進みます。
5. 「バケットアクセス許可」バーで、COSにアクセスするアカウントにアクセス権限が設定されているかどうかをチェックします。
 - 「はい」の場合は、次の手順に進んでください。
 - 「いいえ」の場合は**ユーザーを追加**をクリックし、COSにアクセスするアカウントに必要な権限を設定してください。
6. アクセス権限を設定するアカウントに必要な権限があるかどうかをチェックします。
 - 「はい」の場合は、次の手順に進んでください。
 - 「いいえ」の場合は**編集**をクリックし、再設定してください。
7. 「Policy権限設定」バーで、COSにアクセスするアカウントに**policy**権限承認ポリシーが設定されているかどうかをチェックします。

注意：

- バケットのアクセス権限がプライベート読み取り/書き込みであり、かつPolicy権限が匿名アクセスの場合、Policy権限の優先順位はバケットのアクセス権限より高くなります。

- Policy権限承認ポリシーにおいて、同一のサブユーザーが許可と禁止のポリシーを同時に設定した場合、禁止ポリシーの優先順位は許可ポリシーより高くなります。
- Policy権限承認ポリシーにおいて、「全ユーザー」ポリシーの優先順位は「指定ユーザー」ポリシーより低くなります。

- 「はい」の場合は、次の手順に進んでください。
- 「いいえ」の場合は**ポリシーの追加**をクリックし、実際の署名アクセス時のアカウントに必要な権限を設定してください。

8. Policy権限を設定するアカウントに必要な権限があるかどうかをチェックします。

- 「はい」の場合は、次の手順に進んでください。
- 「いいえ」の場合は**編集**をクリックし、再設定してください。

9. COSリソースへのアクセス時に使用するq-akパラメータがターゲットバケットに所属するアカウントかどうかをチェックします（大文字と小文字を区別する）。

- 「はい」の場合は、次の手順に進んでください。
- 「いいえ」の場合は、q-akパラメータを対応するターゲットバケットに所属するアカウントに変更してください。

10. COSリソースにアクセスする際にクロスアカウントアクセスではないかを確認します。

- 「はい」の場合は、このアカウントにクロスアカウント権限の承認を行ってください。詳細な操作については、「クロスアカウントのサブアカウントに対し指定ファイルの読み取り/書き込み権限を承認する」をご参照ください。
- 「いいえ」の場合は、[お問い合わせ](#)ください。

Messageが「AccessForbidden」の場合

COSへのアクセスの際に次のメッセージが表示された場合、

```
<Code>AccessDenied</Code>  
<Message>AccessForbidden</Message>
```

次の操作を実行する必要があります。

1. [COSコンソール](#)にログインします。
2. 左側ナビゲーションバーで**バケットリスト**を選択し、バケット管理ページに進みます。

- 操作したいバケットを見つけ、そのバケット名をクリックし、バケット設定ページに進みます。
- 左側ナビゲーションバーで**セキュリティ管理**>**クロスドメインアクセスCORS設定**を選択し、クロスドメインアクセスCORS設定ページに進みます。
- 「クロスドメインアクセスCORS設定」バーで、クロスドメインリクエストかどうかをチェックします。
 - 「はい」の場合は、次の手順に進んでください。
 - 「いいえ」の場合はルールを変更してください。
- 次のコマンドを実行し、クロスドメインリクエストの設定が正しいかどうかをチェックします。

```
curl 'http://bucket-appid.cos.ap-guangzhou.myqcloud.com/object' -voa /dev/null -H 'Origin: クロスドメインアクセスCORS設定のソースOrigin'
```

次のような情報が返されれば、正しく設定されています。

```
(base) [root@VM_32_5_centos ~]# curl 'http://[redacted].cos.ap-guangzhou.myqcloud.com/1.gif' -voa /dev/null -H 'Origin: http://[redacted]'
```

% Total	% Received	% Xferd	Average Speed	Time	Time	Time	Current
			Dload	Upload	Total	Spent	Left
0	0	0	0	0	0	----	0*

```
Trying [redacted]:80...
* Connected to [redacted].cos.ap-guangzhou.myqcloud.com ([redacted] port 80 (#0))
> GET /1.gif HTTP/1.1
> Host: [redacted].cos.ap-guangzhou.myqcloud.com
> User-Agent: curl/7.29.0
> Accept: */*
> Origin: http://[redacted]
>
* Mark bundle as not supporting multiuse
< HTTP/1.1 200 OK
< Content-Type: image/gif
< Content-Length: 307293
< Connection: keep-alive
< Accept-Ranges: bytes
< Access-Control-Allow-Credentials: true
< Access-Control-Allow-Methods: PUT,GET,POST
< Access-Control-Allow-Origin: http://[redacted]
< Access-Control-Expose-Headers: myheader
< Date: Tue, 30 Mar 2021 05:07:44 GMT
< ETag: "2c1f37c1cd1eb5da9a7df5774ed27a53"
< Last-Modified: Tue, 16 Mar 2021 14:03:51 GMT
< Server: tencent-cos
< x-cos-hash-crc64ecma: 63978585-[redacted]1
< x-cos-replication-status: Replica
< x-cos-request-id: NjA2Mjk2MDBfN2ExZ-[redacted]4YTVm
< x-cos-version-id: MTg0NDUxI-[redacted]zA
<
{ [data not shown]
100 300k 100 300k 0 0 783k 0 ---- 783k
* Connection #0 to host [redacted].cos.ap-guangzhou.myqcloud.com left intact
* Closing connection -1
curl: (3) URL using bad/illegal format or missing URL
```

Messageが「You are denied by bucket referer rule」の場合

COSへのアクセスの際に次のメッセージが表示された場合、

```
<Code>AccessDenied</Code>
<Message>You are denied by bucket referer rule</Message>
```

次の操作を実行する必要があります。

- COSコンソールにログインします。

- 「はい」の場合は、次の手順に進んでください。
- 「いいえ」の場合は、q-akパラメータを変更してください。キーのSecretIdはq-akパラメータと一致し、なおかつ大文字と小文字が区別されている必要があります。

2. [APIキー管理](#)に進み、APIキーが有効になっているかどうかをチェックします。

- 「はい」の場合は、[お問い合わせ](#)ください。
- 「いいえ」の場合は、そのAPIキーを有効化してください。

Messageが「InvalidObjectState」の場合

COSへのアクセスの際に次のメッセージが表示された場合、

```
<Code>AccessDenied</Code>
<Message>InvalidObjectState</Message>
```

次の操作を実行する必要があります。

リクエストされたオブジェクトがアーカイブタイプまたはディープアーカイブタイプかどうかをチェックします。

- 「はい」の場合はオブジェクトを復元し、再度アクセスしてください。詳細な操作については、[POST Object restore](#)をご参照ください。
- 「いいえ」の場合は、[お問い合わせ](#)ください。

Messageが「RequestTimeTooSkewed」の場合

COSへのアクセスの際に次のメッセージが表示された場合、

```
<Code>AccessDenied</Code>
<Message>RequestTimeTooSkewed</Message>
```

次の操作を実行する必要があります。

1. OSのタイプに基づいて、クライアントの現在時刻を確認します。



- Windowsシステム（Windows Server 2012の場合）：[> コントロールパネル>時計、言語および地域>日付と時刻の設定。](#)

- Linuxシステム： `date -R` コマンドを実行します。

```
(base) [root@VM_32_5_centos ~]# date -R
Tue, 30 Mar 2021 11:44:00 +0800
```

2. クライアントの現在時刻とサーバーの時刻にずれが存在するか（時刻のずれが15分を超えているか）を判断します。

- 「はい」の場合は時刻を同期してください。
- 「いいえ」の場合は、[お問い合わせ](#)ください。

Messageが「Request has expired」の場合

COSへのアクセスの際に次のメッセージが表示された場合、

```
<Code>AccessDenied</Code>
<Message>Request has expired</Message>
```

次の原因が考えられます。

- リクエストの送信時間が署名の有効期間を過ぎている。
- ローカルのシステム時刻が所在リージョンの時刻と一致していない。

署名の有効期間を再設定するか、またはローカルのシステム時刻を同期する必要があります。それでも解決しない場合は、[お問い合わせ](#)ください。

Messageが「SignatureDoesNotMatch」の場合

COSへのアクセスの際に次のメッセージが表示された場合、

```
<Code>AccessDenied</Code>
<Message>SignatureDoesNotMatch</Message>
```

次の操作を実行する必要があります。

クライアントが計算した署名とサーバーが計算した署名が一致しているかどうかをチェックします。

- 「はい」の場合は、[お問い合わせ](#)ください。
- 「いいえ」の場合は、[リクエスト署名](#)のドキュメントを参照するとともに、COS署名ツールを使用して、実現した署名プロセスをチェックしてください。

リソースアクセス異常 ファイルへのアクセス時に404 NoSuchKey が返される

最終更新日：：2022-04-26 16:54:22

故障について

ファイルにアクセスした際、ファイルが見つからない、またはファイルを表示できないと表示され、404 NoSuchKeyエラーコードが返されます。

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<Error>
  <Code>NoSuchKey</Code>
  <Message>The specified key does not exist.</Message>
  <Resource>cos.ap-guangzhou.myqcloud.com/3.gif</Resource>
  <RequestId>NjA4MjM4NjRFRjM...</RequestId>
  <TraceId>0GvYzZiMmQzYjA2OWYhODk0NTRkMTBiOWVmMDAxODc0OWRkZjk0ZDMLNmI1M2E2MTRlY2MzZDhmNmI5...</TraceId>
</Error>
```

考えられる原因

- アクセスしたファイルパスに入力ミスがある。
- アクセスしたファイルパスの大文字と小文字が一致しない。

処理手順

アクセスしたファイルパスおよび大文字と小文字が正しいかどうかチェックします。

- 「はい」の場合は、[お問い合わせ](#)ください。
- 「いいえ」の場合はファイルパスを変更してください。オブジェクト（Object）の命名ルールについては、[オブジェクトの概要](#)をご参照ください。

ビデオへのアクセス時に正常に再生できない

最終更新日： : 2022-04-26 16:55:19

故障について

ビデオ再生の際、正常に再生できません。例えば、ビデオが音声のみで画面が出ない、画面のみで音声が出ない、再生自体ができないなどの問題です。

考えられる原因

- 元のビデオのコーデック形式に問題がある。
- 元のビデオが破損しているために、ビデオのアップロード後に、音声のみまたは画面のみ、再生自体ができないなどの問題が発生する。

処理手順

1. 元のビデオのコーデック形式が正しいかどうかをチェックします。

- 「はい」の場合は、次の手順に進んでください。
- 「いいえ」の場合は、正しいコーデック形式のビデオをアップロードしてください。具体的な要件については、[仕様と制限](#)をご参照ください。

2. VLCなどの専用プレーヤーを使用して、元のビデオが正常に再生可能かどうかをチェックします。

- 「はい」の場合は、[お問い合わせ](#)ください。
- 「いいえ」の場合は、Tencent Cloudのトランスコード製品を使用して元のビデオの修復を試みることをお勧めします。詳細については、[ビデオトランスコード](#)をご参照ください。

同一リンクからアクセスしたファイルの異常

最終更新日：2022-09-15 14:42:19

故障について

- **現象1**：同一のリンクにアクセスしても、アクセス先のファイルが異なる。
- ****現象2****：ファイルを更新後、アクセス先のファイルが旧ファイルのままである。

考えられる原因

- CDN（Content Delivery Network）のキャッシュが期限切れになっていない。
- ブラウザがローカルキャッシュを無効にしていない。
- アクセス先のファイルがハイジャックされ、アクセスしたリソースのコンテンツが想定したコンテンツと一致しなくなっている。

処理手順

CDNキャッシュが期限切れになっているかチェックする

[CDNキャッシュの設定](#)のよくあるご質問のドキュメントを参照し、CDNキャッシュが期限切れとなっているかを判断します。

- 「はい」の場合は、[ブラウザがローカルキャッシュを無効にしているかどうかをチェックする](#)を行ってください。
- 「いいえ」の場合は、[キャッシュ更新](#)のドキュメントを参照し、CDN URLまたはCDNディレクトリを更新してください。

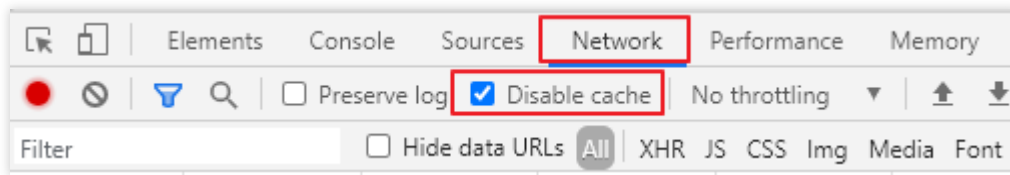
ブラウザがローカルキャッシュを無効にしているかどうかをチェックする

説明：

次の操作はGoogleブラウザの例です。

1. Googleブラウザを開きます。
2. **F12**を押し、デバックウィンドウを開きます。

3. **Network**タブを選択し、**Disable cache**にチェックが入っているかどうかを確認します。



- 「はい」の場合は、[アクセスしたファイルがハイジャックされていないかをチェックする](#)を行ってください。
- いいえの場合は、**Disable cache**にチェックを入れ、ブラウザを再起動してください。

アクセスしたファイルがハイジャックされていないかをチェックする

アクセスしたリソースのコンテンツが想定したコンテンツと一致しない（例えば、ファイルのcontent-lengthが異なる、レスポンスのheaderが異なるなど）場合は、ハイジャックされています。HTTPSプロトコルを使用してこのファイルにアクセスすることをお勧めします。

CDNドメイン名を使用したCOSへのアクセス時にHTTP ERROR 403が返される

最終更新日：：2022-04-26 16:56:56

故障について

CDN（Content Delivery Network）ドメイン名を使用してCOS（Cloud Object Storage）にアクセスすると、エラーコードHTTP ERROR 403が返されます。

考えられる原因

CDNアクセラレーションドメイン名が無効になっている。

処理手順

1. [COSコンソール](#)にログインします。
2. 左側ナビゲーションバーで[バケットリスト](#)を選択し、バケット管理ページに進みます。
3. 操作したいバケットを見つけ、そのバケット名をクリックし、バケット設定ページに進みます。
4. 左側ナビゲーションバーで[ドメイン名と伝送管理](#)>[デフォルトCDNアクセラレーションドメイン名](#)を選択し、デフォルトCDNアクセラレーションドメイン名のページに進みます。
5. 「デフォルトCDNアクセラレーションドメイン名」バーで、現在のステータスが無効な状態になっていないかをチェックします。
 - 無効になっている場合は、[デフォルトのCDNアクセラレーションドメイン名の有効化](#)を行ってください。
 - そうでない場合は、次の手順に進みます。
6. 「カスタムCDNアクセラレーションドメイン名」バーで、ステータスがリリース済みかどうかをチェックします。
 - 「はい」の場合は、[お問い合わせ](#)ください。
 - 「いいえ」の場合は、[カスタムCDNアクセラレーションドメイン名の有効化](#)を行ってください。

HTTPプロトコルを使用してカスタムオリジンサーバードメイン名にアクセスした際にエラーが発生する

最終更新日：：2022-04-26 16:57:33

故障について

HTTPSプロトコルを使用してカスタムオリジンサーバードメイン名にアクセスした際にエラーが発生しました。

考えられる原因

証明書の設定ミス、またはカスタムオリジンサーバードメイン名が設定されていない。

処理手順

CDN証明書の使用

1. [CDNコンソール](#)にログインします。
2. 左側ナビゲーションバーで【ドメイン名管理】を選択し、ドメイン名管理ページに進みます。
3. 設定したいドメイン名をクリックし、【HTTPS設定】タブを選択し、HTTPS設定ページに進みます。
4. 「HTTPS設定」バーで【設定に進む】をクリックし、証明書の設定を行います。
詳細については、CDNの[証明書の設定](#)のドキュメントをご参照ください。
5. 約5分後にCDNドメイン名の再デプロイが完了すると、HTTPSプロトコルによるアクセスが正常に行えるようになります。

COS証明書の使用

1. [COSコンソール](#)にログインします。
2. 左側ナビゲーションバーで【バケットリスト】を選択し、バケット管理ページに進みます。
3. 操作したいバケットを見つけ、そのバケット名をクリックし、バケット設定ページに進みます。

4. 左側ナビゲーションバーで【ドメイン名と伝送管理】>【カスタムオリジンサーバードメイン名】を選択し、カスタムオリジンサーバードメイン名のページに進みます。
5. 操作したいドメイン名を選択し、【証明書のバインド】をクリックし、証明書の設定を行います。

注意：

証明書バインド機能は段階的にリリース中です。現時点では北京、上海、成都、中国香港リージョンのみサポートしています。ワイルドカードドメイン名のバインドはサポートしていません。

6. 【OK】をクリックして、バインドを完了します。

「HTTPS証明書」が「アップロード済み」となっていれば、HTTPSプロトコルによるアクセスが正常に行えます。

CVMリバースプロキシ証明書の使用

[カスタムドメイン名の設定によるHTTPSアクセスのサポート](#)をご参照ください。

POST Objectの一般的な異常

最終更新日： : 2022-04-26 16:38:16

障害の現象

COS APIを使用してPOSTリクエストを行うと、次のような異常を示すエラーコードが返されます。

- [Condition key q-ak doesn't match the value XXXXXX](#)
- [You post object request has been expired, expiration time: 1621188104 but the time now : 1621245817](#)
- [The Signature you specified is invalid.](#)
- [You must provide condition if you specify a policy in post object request.](#)
- [Condition key bucket doesn't match the value \[bucket-appid\]](#)
- [Condition key key doesn't match the value XXXXX](#)
- [The body of your POST request is not well-formed multipart/form-data.](#)

トラブルシューティング

Messageが「Condition key q-ak doesn't match the value XXXXXX」の場合

COS APIを使用してPOSTリクエストを行うと次のメッセージが表示された場合：

```
<Code>AccessDenied</Code>
<Message>Condition key q-ak doesn't match the value XXXXXX</Message>
```

考えられる原因

q-akパラメータの入力エラー。

ソリューション

1. CAMコンソールにログインし、[【APIキー管理】](#) ページに進み、キー情報を確認します。
 2. 確認したキー情報を基に、q-akパラメータに入力エラーがあるかどうかを確認します。
- 「はい」の場合は、q-akパラメータを正しいSecretIdに変更してください。
 - 「いいえ」の場合は、[お問い合わせ](#)ください。

Messageが「You post object request has been expired, expiration time: 1621188104 but the time now : 1621245817」の場合

COS APIを使用してPOSTリクエストを行うと次のメッセージが表示された場合：

```
<Code>AccessDenied</Code>
<Message>You post object request has been expired, expiration time: 1621188104 but the time now : 1621245817</Message>
```

考えられる原因

Policyの中のexpirationの値が期限切れになっている。

ソリューション

Policyの中のexpirationの値を変更してください。

注意：

expirationの値は現在時刻より後である必要があります。現在時刻+30分（UTC時刻）に設定することをお勧めします。

Messageが「The Signature you specified is invalid.」の場合

COS APIを使用してPOSTリクエストを行うと次のメッセージが表示された場合：

```
<Code>SignatureDoesNotMatch</Code>
<Message>The Signature you specified is invalid.</Message>
```

考えられる原因

署名の計算が間違っている。

ソリューション

[リクエスト署名](#)のドキュメントを参照し、POST署名文字列の生成ルールが正しいかどうかをチェックしてください。

Messageが「You must provide condition if you specify a policy in post object request.」の場合

COS APIを使用してPOSTリクエストを行うと次のメッセージが表示された場合：

```
<Code>InvalidPolicyDocument</Code>
<Message>You must provide condition if you specify a policy in post object request.</Message>
```


考えられる原因

Policyの形式が間違っている。

ソリューション

[POST Object](#)のドキュメントを参照し、Policyの形式を標準のJSON形式に変更してください。

Messageが「Condition key bucket doesn't match the value [bucket-appid]」の場合

COS APIを使用してPOSTリクエストを行うと次のメッセージが表示された場合：

```
<Code>AccessDenied</Code>
<Message>Condition key bucket doesn't match the value [bucket-appid]</Message>
```

考えられる原因

Policyのbucketとリクエストされたbucketが異なる。

ソリューション

Policyのbucketを使用してリクエストを行ってください。

Messageが「Condition key key doesn't match the value XXXXX」の場合

COS APIを使用してPOSTリクエストを行うと次のメッセージが表示された場合：

```
<Code>AccessDenied</Code>
<Message>Condition key key doesn't match the value XXXXX</Message>
```

考えられる原因

アップロードしたコンテンツがpolicyルールに適合していない。

ソリューション

PolicyのConditionに基づき、その条件に適合したコンテンツをアップロードします。

Messageが「The body of your POST request is not well-formed multipart/form-data.」の場合

COS APIを使用してPOSTリクエストを行うと次のメッセージが表示された場合：

```
<Code>MalformedPOSTRequest</Code>
<Message>The body of your POST request is not well-formed multipart/form-data.</Message>
```

考えられる原因

POST bodyの形式が規定に適合していない。

ソリューション

[POST Object](#)のドキュメントを参照し、bodyの形式を最適化します。