



## 2018 ベリトランスサーバー証明書更新

### 接続検証手順

～VeriTrans3G/VeriTrans4G(MDK および MDKトークン型)～

Ver. 1.0 (2018年2月)

## 目次

1.	接続検証の手順.....	3
1.1.	マーチャント様環境の準備.....	3
1.2.	検証環境への接続設定.....	3
1.3.	検証用取引の実行.....	3
1.4.	検証結果の確認.....	4
1.5.	検証結果のご連絡.....	4
2.	通信エラー時の対応について.....	5
2.1.	ルート証明書の確認方法.....	5
2.1.1.	Java をご利用の場合.....	5
2.1.2.	.NET をご利用の場合.....	5
2.1.3.	PHP をご利用の場合.....	6
2.1.4.	Ruby をご利用の場合.....	6
3.	本番運用における注意点.....	6
4.	その他.....	7
4.1.	著作権、および問合せ先.....	7
4.2.	改定履歴.....	7

# 1. 接続検証の手順

MDK を利用している全てのシステムで、接続検証を行って頂きますようお願いいたします。

検証を行って頂く際は、本番サーバと同等スペックのマーチャント様環境から疎通確認を行って頂いたうえで、最終確認として本番環境からの接続検証を行って頂くことを推奨いたします。

## 1.1. マーチャント様環境の準備

マーチャント様が検証を行う環境は、本番サーバと同等の環境のご用意をお願いいたします。

また、最終確認のために本番サーバでの検証も行って頂きますようお願いいたします。

- ✓ OS、ミドルウェア、プログラミング言語環境のバージョン等が同一のサーバをご用意ください。特に、SSL 通信に関連する設定が本番と同じであることをご確認ください。
- ✓ マーチャント様の検証サーバ環境では発生しなかった問題が、本番サーバで発生する可能性もありますので、本番サーバでも接続検証を実施してください。

## 1.2. 検証環境への接続設定

MDK 設定ファイルの接続先 URL (HOST\_URL) を、ベリトランス検証環境の URL に変更してください。

下表に、各言語の設定ファイル名と変更箇所を示します。

開発言語	MDK 設定ファイル名	接続先 URL の変更
Java	3GPSMDK.properties	変更前(本番環境 URL)
PHP	3GPSMDK.properties	HOST_URL = <a href="https://api.veritrans.co.jp:443">https://api.veritrans.co.jp:443</a>
.NET	3GPSMDK.ini	変更後(検証環境 URL)
Ruby	tg_mdk.ini	HOST_URL = <a href="https://sandbox.api.veritrans.co.jp:443">https://sandbox.api.veritrans.co.jp:443</a>

※ 必要に応じてアプリケーションサーバを再起動し、MDK の設定を反映して下さい。

## 1.3. 検証用取引の実行

ベリトランス検証環境に取引要求を送信してください。

- ✓ SSL 通信が成功することの確認が目的のため、取引要求の種類は問いません。
- ✓ 本番モード(DUMMY\_REQUEST=0)、ダミーモード(DUMMY\_REQUEST=1)のどちらのモードでも実行可能です。
  - ◇ ベリトランス検証環境は、本番取引が実行できるように設定しています。そのため、コマンド(機能)によっては実際に与信枠が確保され、売上が成立しますので、必ず取消を行って頂きますようお願いいたします。
- ✓ 検証に利用するマーチャント ID は、現在ご利用の本番用マーチャント ID をご利用頂きますようお願いいたします。
  - ◇ 本番用マーチャント ID のご利用が難しい場合は、全マーチャント様で共用のマーチャント ID をご利用頂くことも可能です。共用のマーチャント ID では、全ての取引がダミーモードとなりますのでご留意下さい。
  - ◇ 共用のマーチャント ID は、ベリトランスのサポートサイトより取得可能です。

<https://www.veritrans.co.jp/trial/login/>

## 1.4. 検証結果の確認

取引要求がベリトランス検証環境に向けて送信されており、SSL 通信が正常に行われたことをご確認ください。

- ✓ MDK のログに出力されている接続先 URL が、ベリトランス検証環境の URL であることをご確認ください。
  - ◇ MDK のログに、以下のように URL が出力されていることをご確認ください。  
`connect url ==> https://sandbox.api.veritrans.co.jp:443/*****`
- ✓ 取引要求に対し、ベリトランス検証環境の決済サーバからの応答が正常に受信できることをご確認ください。
  - ◇ エラーが発生した場合、結果コード (VResultCode) が「決済サーバが返すコード (M で始まるコード以外のコード)」であれば、SSL 通信が確立できているためテストは成功です。
  - ◇ 結果コード (VResultCode) の先頭が M で始まるエラー (MF02/MF03/MF04/MF99 etc.) が発生した場合は、検証環境との接続に失敗しています。
  - ◇ 接続に失敗した場合は、MDK の出力ログをご確認ください。ベリトランスの新しい証明書を認識できないことが原因の場合は、ルート証明書の追加等の対応が必要です。詳しくは「2.通信エラー時の対応について」をご参照ください。
  - ◇ OS やミドルウェアのログも合わせてご確認ください。例えば、PHP の場合は WEB サーバ (apache のログ等) に何らかのエラーが出力されている場合がございます。

## 1.5. 検証結果のご連絡

検証の状況を弊社でも確認させていただきますので、検証が終わりましたら、以下の方法で結果をご連絡いただきますよう、お願いいたします。

- ✓ 検証結果送信用フォームの URL を別途メールにてお知らせいたしますので、その URL にアクセスし、画面に表示される指示の通りに情報を入力して最後に送信ボタンを押してください。
  - ◇ 以下、ご入力いただく項目の一部になります。
    - ・ 本番用マーチャント ID / 検証時マーチャント ID
    - ・ 検証実施担当者の氏名、メールアドレス
    - ・ 検証時サーバの接続元グローバル IP アドレス
    - ・ 取引 ID
    - etc.

## 2. 通信エラー時の対応について

通信エラーが発生した場合、最初に以下の3点についてご確認ください。

- ✓ 検証環境に接続するためのシステム要件(TLS1.1/1.2 での通信)に対応しているか
- ✓ 下表に示すベリトランス検証環境との通信が、マーチャント様のネットワークで許可されているか

IP アドレス	ホスト名
210.239.44.181	sandbox.api.veritrans.co.jp

- ✓ 「1.2 検証環境への接続設定」に記載のとおり正しく設定を行っているか。

これらの点に問題がない場合、ベリトランスサーバー証明書のルート証明書がシステムに存在しない可能性があります。ご利用のプログラミング言語毎の対応方法を以下にご説明します。

### 2.1. ルート証明書の確認方法

#### 2.1.1. Java をご利用の場合

MDK の設定ファイルに指定されている cacerts ファイルを、MDK に同梱の最新の cacerts に更新してください。

独自の cacerts をご利用の場合は、以下のサイトよりルート証明書をダウンロードし、keytool コマンドで、cacerts にインポートしてください。

- ルート証明書(DigiCert High Assurance EV Root CA)のダウンロード URL  
<https://www.digicert.com/CACerts/DigiCertHighAssuranceEVRootCA.crt>
- keytool コマンドの利用方法

```
keytool -import -alias digicerthighassuranceevrootca -keystore cacerts -file DigiCertHighAssuranceEVRootCA.crt
```

#### 2.1.2. .NET をご利用の場合

Windows のルート証明書ストアに、以下のルート証明書が格納されていない場合は、WindowsUpdate を行っていただくか、手動でインポートして下さい。

- ルート証明書(DigiCert High Assurance EV Root CA)のダウンロード URL  
<https://www.digicert.com/CACerts/DigiCertHighAssuranceEVRootCA.crt>

ルート証明書のインポートの後、以下の URL にブラウザからアクセスし、「DigiCert」のページが表示されればインポートは成功です。

```
https://global-root-ca.chain-demos.digicert.com/
```

### 2.1.3. PHP をご利用の場合

MDK の設定ファイルに指定されている cert.pem ファイルを、MDK に同梱の最新の cert.pem ファイルに更新してください。

### 2.1.4. Ruby をご利用の場合

MDK の設定ファイルに指定されている cert.pem ファイルを、MDK に同梱の最新の cert.pem ファイルに更新してください。

**※以上の対応を行っても通信エラーが改善されない場合は、ベリトランステクニカル・サポートまで MDK のログを添付してお問い合わせください。**

## 3. 本番運用における注意点

接続検証完了後の本番運用におかれましては、以下の点にご注意ください。

- ✓ 「1.2 検証環境への接続設定」で実施した接続先 URL の設定変更は、必ず元に戻してください。
- ✓ 通信エラー発生時に実施した 1.2 以外のシステム変更につきましては、本番環境にも適用が必要です。
  - 本番サーバで検証を行った場合は、接続先 URL 以外はそのままの設定で本番運用を行ってください。
  - 本番サーバで検証を行わなかった場合は、接続先 URL 変更以外のシステム変更について、必ず本番サーバにも適用してください。

## 4. その他

### 4.1. 著作権、および問合せ先

[著作権] 本ドキュメントの著作権はベリトランス株式会社が保有しています。

Copyright (c) 2018 VeriTrans Inc., a Digital Garage company. All rights reserved.

[お問い合わせ先] ベリトランス株式会社 テクニカルサポート SSL-SHA2 関連窓口

電子メール: ssl-sha2@veritrans.jp

### 4.2. 改定履歴

2018/2/22 : Ver1.0 リリース