# 旧: サーバ証明書の設定(IdPv3)

## サーバ証明書の取得とApacheの設定

１．「UPKI電子証明書発行サービス」の利用管理者編をご覧いただき、サーバ証明書発行を申請します。機関の審査手続きによっては証明書の交付までには数日を要する場合がありますので、お早めに申請してください。
接続実験をするだけであれば、IdPインストール時に作成された証明書（自己署名証明書）をそのまま利用してテストフェデレーションに参加することも可能です。その場合は、以降の記述のうち「中間CA証明書」の部分は無視してください。

２．入手したサーバ証明書をもとに、以下のファイルに設定してください。

### ■/etc/httpd/conf.d/ssl.conf

まず、秘密鍵を"root"ユーザのみが参照できるようにアクセス制限がかかっているか確認してください。確認できない場合は以下のようにして所有者・グループ・パーミッションを設定してください。

```
chown root:root /etc/pki/tls/private/server.key       ← 秘密鍵の格納先
chmod 400 /etc/pki/tls/private/server.key
```

/etc/httpd/conf.d/ssl.conf を以下のように編集してください。

```
（省略）
SSLCertificateFile /etc/pki/tls/certs/server.crt            ← サーバ証明書の格納先
（省略）
SSLCertificateKeyFile /etc/pki/tls/private/server.key       ← 秘密鍵の格納先
（省略）
SSLCertificateChainFile /etc/pki/tls/certs/server-chain.crt ← 中間CA証明書の格納先
↑ 先頭の「#」を削除して、コメントを解除してください。
```

※端末のサイズによっては表記がずれる可能性がございます。画面を広くしてご覧ください。

詳しくは、サーバ証明書インストールマニュアルの Apache 2 + mod_ssl 編を参照してください。

---

## メタデータの作成と提出

学認申請システム（テストFed）から登録を行います。入力したデータから自動的にメタデータが生成されますのでメタデータ作成の必要はありません。詳細は参加のページをご覧ください。
⇒参加
学認申請システムから登録を行います。入力したデータから自動的にメタデータが生成されますのでメタデータ作成の必要はありません。詳細は参加のページをご覧ください。
⇒参加

---

## Back-Channelの設定

SAML 1のSPにも接続する場合は、IdPとの通信時にTLS接続を行うため、下記にしたがいBack-Channelの設定を行ってください。 このTLS接続ではポート8443を利用します。

### １．キーストアの設定

サーバ証明書を格納したキーストアを作成します。

```
# cd /opt/shibboleth-idp/credentials
# UMASKORIG=”`umask`” ; umask 0077
# openssl pkcs12 -export -out server.p12 -in サーバ証明書.crt -inkey サーバ秘密鍵.key -name サーバ名
（ここで聞かれるエクスポートパスワードを後述のserver.xmlの「P12パスワード」に指定します。任意のものを設定できます）
# umask ”$UMASKORIG”
（上記一連のumaskコマンドは”chmod 600 server.p12”と同義）
```

Tomcatを"tomcat"ユーザで実行する場合は、さらに以下のコマンドを実行しTomcatが読み取れるようにします。

```
# chgrp tomcat /opt/shibboleth-idp/credentials/server.p12
# chmod g+r /opt/shibboleth-idp/credentials/server.p12
```

## 2．ライブラリのコピー

https://build.shibboleth.net/nexus/service/local/repositories/releases/content/net/shibboleth/utilities/trustany-ssl/1.0.0/trustany-ssl-1.0.0.jar
よりダウンロードします。
trustany-ssl-1.0.0.jar を $CATALINA_HOME/lib 配下にコピーします。

```
# wget https://build.shibboleth.net/nexus/service/local/repositories/releases/content/net/shibboleth/utilities/trustany-ssl/1.0.0
/trustany-ssl-1.0.0.jar
# cp trustany-ssl-1.0.0.jar $CATALINA_HOME/lib/
```

ダウンロードされるJARファイルのSHA-256ハッシュ値は以下の通りです。さらに真正性を確認したい場合はPGP署名をご利用ください。

```
# sha256sum trustany-ssl-1.0.0.jar
80f80f45dcb6671ad963e6ebf761baeb195502ca5b274b6b3f99e4ed2a900466  trustany-ssl-1.0.0.jar
```

## 3．SOAP設定

$CATALINA_BASE/conf/server.xml ファイルに以下を追加します。

```
<Connector port=”8443” protocol=”org.apache.coyote.http11.Http11NioProtocol”
        maxThreads=”150”
        SSLEnabled=”true”
        scheme=”https”
        maxPostSize=”100000”
        secure=”true”
        clientAuth=”want”
        sslEnabledProtocols=”TLSv1,TLSv1.1,TLSv1.2,TLSv1.3”
        keystoreFile=”/opt/shibboleth-idp/credentials/server.p12”
        keystorePass=”P12パスワード”
        keystoreType=”PKCS12”
        trustManagerClassName=”net.shibboleth.utilities.ssl.TrustAnyCertificate” />
```

sslEnabledProtocols はSSLv3を無効にするための記述です。なお、TLSv1.1、TLSv1.2、およびTLSv1.3の記載がありますが、実際に使用できるか否かはJava(JVM)のバージョンに依存します。