JCCH・セキュリティ・ソリューション・システムズ

プライベートCA Gléas ホワイトペーパー

OpenCA-OCSPDでのOCSPレスポンダー設定手順

Ver.1.0

2015 年 6 月

目次

1. はじめに

1.1. 本書について

本書では、弊社製品「プライベートCA Gléas」で発行した失効リストを利用して、OCSP（Online Certificate Status Protocol）を用いて失効確認をおこなう環境を構築するための設定例を記載します。

本書に記載の内容は、弊社の検証環境における動作を確認したものであり、あらゆる環境での動作を保証するものではありません。弊社製品を用いたシステム構築の一例としてご活用いただけますようお願いいたします。

弊社では試験用の証明書類の提供も行っております。検証等で必要な場合は、最終項のお問い合わせ先までお気軽にご連絡ください。

1.2. 本書における環境

本書における手順は、以下の環境で動作確認を行っています。

- OCSPレスポンダー：
  Ubuntu Server 14.04 LTS
  OpenCA-OCSPD 3.1.1 / LibPKI 0.8.8
  （CRL取得用スクリプト copycrl.rb 動作用）Ruby 1.9.3p484
  ※以後、「OCSPレスポンダー」と記載します
- JS3 プライベートCA Gléas （バージョン1.12）
  ※以後、「Gléas」と記載します
- クライアント：Ubuntu Server 14.04 LTS
  CUIクライアント： OpenSSL 1.0.1f
  GUIクライアント： Apache/2.4.7 (Ubuntu) / mod_ssl
  （Webクライアント）Windows8.1 Pro / Internet Explorer 11

以下については、本書では説明を割愛します。

- Ubuntu Serverのセットアップ
- Gléasでのユーザ登録やクライアント証明書発行等の基本設定
- Apacheのセットアップや設定

これらについては、各製品のマニュアルをご参照いただくか、各製品を取り扱っている販売店にお問い合わせください。

1.3. 【ご参考】インストール手順

本書での環境におけるOCSPレスポンダーのインストール手順を記載します。
LibPKIやOCSPDのtar.gzパッケージはホームディレクトリに事前にダウンロードされているものとし、また以下手順に含まれるコマンドもインストールされているものとします。

【LibPKIのインストール】

```
$ cd ~
$ sudo apt-get install libldap2-dev libxml2-dev libssl-dev
$ tar zxvf libpki-0.8.8.tar.gz
$ cd libpki
$ ./configure
$ make
$ sudo make install
```

【OpenCA-OCSPDのインストール】

```
$ cd ~
$ tar zxvf openca-ocspd-3.1.1.tar.gz
$ cd openca-ocspd
$ ./configure
$ make
$ sudo make install
```

【LibPKIライブラリへパスを通す】

```
$ sudo sh -c "echo /usr/lib64/ > /etc/ld.so.conf.d/ ¥
ocspd.conf"
$ sudo ldconfig
```

【起動スクリプトを /etc/init.d/ にリンクさせる】

```
$ sudo ln -s /usr/etc/init.d/ocspd /etc/init.d/ocspd
```

2. OCSPレスポンダーの設定

2.1. OCSP 証明書の配置

次のファイルを OCSP レスポンダーにアップロードします。
ファイル名、アップロード先ディレクトリ共にサンプルとなります。本書では以下の名前であることを前提に記載します。
※各ファイル名は以下ファイルの設定どおりにします。
/usr/etc/ocspd/pki/token.d/software.xml

	ファイル名	アップロード先ディレクトリ名
CA 証明書	cacert.pem	/usr/etc/ocspd/certs/
OCSP 証明書	cert.pem	/usr/etc/ocspd/certs/
OCSP 秘密鍵	key.pem	/usr/etc/ocspd/private/
証明書失効リスト（CRL）	crl_ia1.pem	/usr/etc/ocspd/crls/

【CA 証明書】
Gléas では次の URL から取得します。
http://{Gléas のホスト名 or IP アドレス}/crl/ia1.pem
ファイル名を上記のものに変更し、上記ディレクトリに配置します。

【OCSP 証明書・秘密鍵】
Gléas からダウンロードした OCSP 証明書は PKCS#12 という形式になっているため、PEM 形式に変換・分離する必要があります。
※ここでは Gléas より取得したファイルを ocspd.p12 というファイル名と仮定します。
1) PKCS#12 ファイルより証明書を取得

```
$ openssl pkcs12 -in ocspd.p12 -clcerts -nokeys -out cert.pem
```

2) PKCS#12 ファイルより秘密鍵を取得

```
$ openssl pkcs12 -in ocspd.p12 -nocerts -nodes -out key.pem
```

取得したファイルを上記ディレクトリに配置します。
また、秘密鍵は ocspd の実行ユーザ（デフォルトでは www-data）をオーナーにし、パーミッションを 400 に変更します。

【証明書失効リスト（CRL）】
Gléas では CRL ファイル（PEM 形式）は次の URL から取得できます。
http://{Gléas ホスト名 or IP アドレス}/crl/crl_ia1.pem

取得したファイルを上記ディレクトリに配置します。

本書の設定では、Gléas で証明書を失効しても OCSP レスポンダー上の CRL が自動的に更新されることはありません。新しい CRL が発行された後には既存の CRL と置き換える必要があります。
Gléas では CRL 更新用の ruby スクリプト（copycrl.rb）を準備しております。
corycrl.rb では、以下のとおりにすることで Gléas より CRL を取得・置換することが可能です。

```
$ ./copycrl.rb [CRL 取得 URL] /usr/etc/ocspd/crls/crl_ia1.pem
```

一定間隔で実行することで CRL を継続的に更新し、2.2 項の crlAutoReload により定期的に失効情報を OCSP レスポンダーに反映させることが可能です。
急を要する場合などは、以下を実行することでも取り込んだ CRL の即時反映が可能です。

```
$ sudo service ocspd reload-crl
```

2.2. 設定ファイルの編集

CA 設定ファイルを作成します。本書環境では以下にあります。
/usr/etc/ocspd/ca.d/
既存の self-certs.xml をコピーして編集します。（既存のファイルは、拡張子を変更するなどで起動時に読み込まれないようにします）

変更点は以下の通りです。（設定ファイルの説明はここではしません）

- <pki:name>タグには任意の名称を設定します。
- <pki:caCertUrl>には、CA 証明書を配置したファイルパスを指定します。
- <pki:crlUrl>には、CRL を配置したファイルパスを指定します。

OCSPD の設定ファイルを編集します。本環境では以下にあります。
/usr/etc/ocspd/ocspd.xml

変更点は以下の通りです。（設定ファイルの説明はここではしません）

- <pki:crlAutoReload>には、CRL のリロードをおこなう秒数を指定します。
- <pki:crlReloadExpired>を yes に設定することで、有効期限が過ぎた CRL をそのまま利用することが可能になります。
  （CRL の安全性レベルは落ちますが、可用性が向上します）

2.3. OCSP レスポンダーの起動

以下コマンドにより起動します。

```
$ sudo service ocspd start
```

※デバッグモードで起動したい場合は以下の通りにします。

ログに詳細なメッセージ出力されます。

```
$ sudo service ocspd start-debug
```

以下にログが記録されます。

/var/log/syslog

自動起動させる場合は、以下をおこないます。

```
$ sudo update-rc.d ocspd default
```

以上で、OCSP レスポンダーの設定は終了です。

3. 動作確認（クライアント証明書認証）

3.1. CUI での確認

OpenSSLを使っての動作確認が可能です。

```
$ openssl ocsp  -issuer [CA証明書] -serial [失効確認する証明書のシリアル番号(10進数か、「0x」を前につけた16進数)] -url [OCSPレスポンダーのURL（ホスト名：ポート番号（デフォルトでは2560））] -VAfile [OCSP証明書] -CAfile [CA証明書]
```

指定した証明書が有効な場合、以下のレスポンスが標準出力に表示されます。

```
Response verify OK
[シリアル番号]:  good
        This Update: Jun xx 00:00:00 2015 GMT
        Next Update: Jun xx 00:05:00 2015 GMT
```

指定した証明書が失効している場合のレスポンスは以下のとおりです。

```
Response verify OK
[シリアル番号]: revoked
        This Update: Jun xx 00:00:00 2015 GMT
```

```
Next Update: Jun xx 00:05:00 2015 GMT
Reason: superseded
Revocation Time: Jun xx 00:00:00 2015 GMT
```

OCSPDに設定されていない認証局から発行された証明書の失効確認をおこなう場合のレスポンスは以下のとおりです。

```
Response verify OK
[シリアル番号]: unknown
        This Update: Jun xx 00:00:00 2015 GMT
        Next Update: Jun xx 00:05:00 2015 GMT
```

引数に `-text` を追加するとOCSPリクエストとレスポンスの詳細が表示されます。

3.2. GUI での確認

※Apacheでのクライアント証明書認証の設定に関しては、以下URLで公開されている弊社ホワイトペーパー「Apache におけるクライアント証明書を利用したユーザ認証」をご参照ください。
http://www.jcch-sss.com/service/support/2010/09/apache-ssl-client-auth
ここでは、OCSPを使った失効確認の設定だけを記載します。

失効確認をおこなうクライアント証明書を発行した認証局と、OCSP証明書を発行した認証局が異なる場合（Gléasの管理用CAでOCSP証明書を発行した場合など）は、証明書の目的（Trusted Uses）を追加してから、SSLCACertificatePathディレクティブで指定するディレクトリに配置しハッシュリンクを作成します。
以下はopensslコマンドを使った例です。

```
$ openssl x509 -in ia2.pem -addtrust OCSPSigning -out ocspCA.pem
```

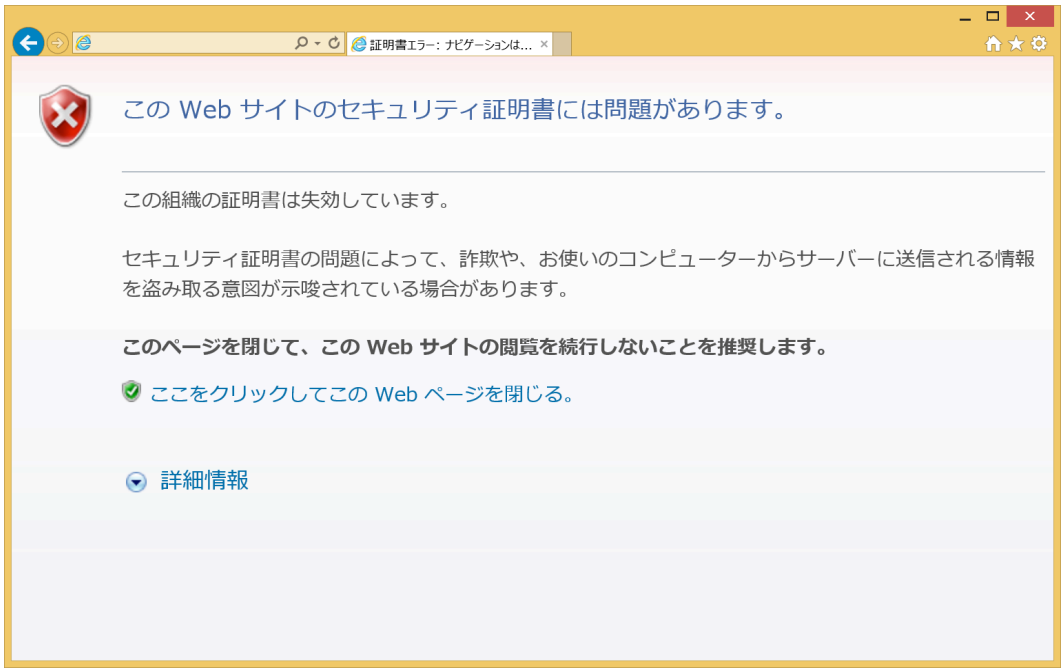
Apacheの設定ファイルを編集します。
※UbuntuのApache2パッケージの場合は、以下のディレクトリに設定ファイルがあります。
/etc/apache2/sites-available/

```
SSLVerifyClient require
SSLVerifyDepth  1
```

上記ディレクティブの下に以下ディレクティブを追加します。

```
SSLOCSPEnable on
SSLOCSPDefaultResponder [OCSPレスポンダーのURL]
SSLOCSPOverrideResponder on
```

設定終了後にApacheを再起動します。
Internet Explorerから失効した証明書を用いて、Webアクセスすると以下の通り表示されます。



Apacheのエラーログには以下が記録されます。

```
[ssl:error] [pid xxxx:tid xxxxxxxxxxxxxxx] [client xxx.xxx.xxx.xxx:xxxxx]
OCSP validation completed, certificate status: revoked (1, -1) [subject:
CN=user / issuer: CN= certificate authority / serial: xx / notbefore: Jun
01 00:00:00 2015 GMT / notafter: Jun 30 23:59:59 2015 GMT]
[ssl:error] [pid xxxx:tid xxxxxxxxxxxxxxx] [client xxx.xxx.xxx.xxx:xxxxx]
AH02039: Certificate Verification: Error (23): certificate revoked
```

4. 動作確認（OCSP Stapling）

GléasでWebサーバ用の証明書を発行している場合、Webサーバ側でOCSP Staplingを設定することで失効確認結果をクライアントに送信することが可能となります。

Apacheの設定ファイルを編集します。
<VirtualHost>セクションの外側に以下ディレクティブを追加します。

```
SSLStaplingCache shmcb:/var/run/ocsp(128000)
```

```
SSLUseStapling on
SSLStaplingResponderTimeout 5
SSLStaplingReturnResponderErrors off
SSLStaplingForceURL [OCSPレスポンダーのURL]
```

設定終了後にApacheを再起動します。
OpenSSLで確認することが可能です。

```
$ openssl s_client -connect localhost:443 -status
```

正常に設定されていると以下のようなレスポンスが返されます。

```
OCSP response:
======================================
OCSP Response Data:
    OCSP Response Status: successful (0x0)
    Response Type: Basic OCSP Response
    Version: 1 (0x0)
    Responder Id: [OCSP証明書のサブジェクト]
    Produced At: Jun 19 xx:xx:xx 2015 GMT
    Responses:
    Certificate ID:
      Hash Algorithm: sha1
      Issuer Name Hash: A83FDEB91BB2627BC1C50DBBC9B3470068387EDF
      Issuer Key Hash: D08E24C392D35887DFAE7C5823D4FCF6B590E1E6
      Serial Number: xx
    Cert Status: good
    This Update: Jun 19 xx:xx:xx 2015 GMT
    Next Update: Jun 19 xx:xx:xx 2015 GMT
```

5. 問い合わせ

ご不明な点がございましたら、以下にお問い合わせください。

■Gléasや検証用の証明書に関するお問い合わせ
株式会社JCCH・セキュリティ・ソリューション・システムズ
Tel:　050-3821-2195
Mail:　sales@jcch-sss.com