IdPのトラストアンカーの確認と必要なCA証明書の導入

現在メタデータリポジトリのサーバ証明書はSHA-1証明書を使用していますが、今後SHA-2証明書に移行する予定です。SHA-2証明書への移 行時にはOSやパッケージのバージョンによって、必要となるCA証明書が入っていない可能性があり、確認・導入の作業が発生する見込みです。

すでにIdPを構築済み・運用中の方は、以下の手順に従って利用しているOS、パッケージのバージョンをご確認ください。

これからIdP構築される方は構築段階で確認・導入しておくことで将来行うべき作業が不要となりますので、今回を機会にご対応ください。

確認方法

- CentOS 6で提供されるOpenJDKパッケージを利用している場合
  CentOS 6で提供されるjava-1.6.0-openjdkパッケージおよびjava-1.7.0-openjdkパッケージのトラストアンカーはca-certificatesパッケージにて提供されます。ca-certificateパッケージが ca-certificates-2013.1.94-65.0.el6.noarch.rpm およびそれ以降のバージョンであることを確認してください。異なる場合には最新版にアップデートしてください。
- CentOS 5で提供されるOpenJDKパッケージを利用している場合
  CentOS 5で提供されるjava-1.6.0-openjdkパッケージおよびjava-1.7.0-openjdkパッケージのトラストアンカーは同じパッケージ内に含まれた形で提供されていますが、対象となるCA証明書が含まれるパッケージが提供されていません。手順に従って必要なCA証明書を導入してください。
- Oracle JDK 6を利用している場合
  Oracle JDK 6 Update 17およびそれ以降をご利用の場合はトラストアンカーに必要なCA証明書が含まれています。Oracle JDK 6 Update 16およびそれ以前のバージョンを利用している場合は最新版へアップデートしていただくか、手順に従って必要なCA証明書を導入してください。
- Oracle JDK 7およびそれ以降を利用している場合
  Oracle JDK 7およびそれ以降のトラストアンカーには必要な証明書が含まれています。そのため、CA証明書の導入は不要です。

CA証明書の導入方法

IdPのトラストアンカーにSECOMのSHA-2 CA証明書を導入する手順について記載します。

証明書の導入

・https://repository.secomtrust.net/SC-Root2/index.htmlからSecurity Communication Root CA2証明書をダウンロードします。ダウンロードしたファイルのファイル名は SCRoot2ca.cer とします。

・ダウンロードしたCA証明書のフィンガープリントが一致しているか下記の値と確認してください。

```
$ openssl x509 -inform DER -in SCRoot2ca.cer -noout -fingerprint -md5
MD5 Fingerprint=6C:39:7D:A4:0E:55:59:B2:3F:D6:41:B1:12:50:DE:43

$ openssl x509 -inform DER -in SCRoot2ca.cer -noout -fingerprint -sha1
SHA1 Fingerprint=5F:3B:8C:F2:F8:10:B3:7D:78:B4:CE:EC:19:19:C3:73:34:B9:C7:74

$ openssl x509 -inform DER -in SCRoot2ca.cer -noout -fingerprint -sha256
SHA256 Fingerprint=51:3B:2C:EC:B8:10:D4:CD:E5:DD:85:39:1A:DF:C6:C2:DD:60:D8:7B:B7:36:D2:B5:21:48:4A:A4:7A:0E:BE:F6
```

・OpenJDKまたはOracle JDKがインストールされていることを確認します。本手順ではOracle JDKに関して/usr/java配下にOracle JDK Update 16がインストールされている前提として説明します。

- OpenJDK 1.6.0の場合

```
$ rpm -q java-1.6.0-openjdk
java-1.6.0-openjdk-1.6.0.33-1.13.5.0.el5_11
```

- OpenJDK 1.7.0の場合

```
$ rpm -q java-1.7.0-openjdk
java-1.7.0-openjdk-1.7.0.71-2.5.3.1.el5_11
```

- Oracle JDKの場合は、インストールパスを確認します。

```
$ ls -l /usr/java/jdk1.6.0_16/
```

・トラストアンカー(cacerts)にSecurity Communication Root CA2 証明書 [SR3.0用]が含まれているか否かを調査します。含まれている場合には、MD5, SHA1のフィンガープリントが表示されます。

- OpenJDK 1.6.0/1.7.0の場合。複数のバージョンを導入している場合にはIdPで利用しているOpenJDKのcacertsをチェックしてください。

```
$ keytool -list -keystore /etc/alternatives/jre/lib/security/cacerts | ¥
    grep -e '6C:39:7D:A4:0E:55:59:B2:3F:D6:41:B1:12:50:DE:43' ¥
         -e '5F:3B:8C:F2:F8:10:B3:7D:78:B4:CE:EC:19:19:C3:73:34:B9:C7:74'

Enter keystore password:   ← パスワードを変更していない場合はデフォルトのパスワード'changeit'を入力するか、

                              空のままEnterを入力します。
```

- Oracle JDKの場合

```
$ /usr/java/jdk1.6.0_16/bin/keytool -list -keystore /usr/java/jdk1.6.0_16/jre/lib/security/cacerts | ¥
    grep -e '6C:39:7D:A4:0E:55:59:B2:3F:D6:41:B1:12:50:DE:43' ¥
         -e '5F:3B:8C:F2:F8:10:B3:7D:78:B4:CE:EC:19:19:C3:73:34:B9:C7:74'
```

・トラストアンカー(cacerts)にSecurity Communication Root CA2 証明書 [SR3.0用]をインポートします。（aliasオプションを付けない場合はmykeyというエイリアス名になります）

- OpenJDKの場合

```
$ sudo keytool -import -alias scrootca2 -keystore /etc/alternatives/jre/lib/security/cacerts -file SCRoot2ca.cer

Enter keystore password:  ← パスワードを変更していない場合はデフォルトの 'changeit' を入力します。

Owner: OU=Security Communication RootCA2, O="SECOM Trust Systems CO.,LTD.", C=JP

Issuer: OU=Security Communication RootCA2, O="SECOM Trust Systems CO.,LTD.", C=JP

Serial number: 0

Valid from: Fri May 29 14:00:39 JST 2009 until: Tue May 29 14:00:39 JST 2029

Certificate fingerprints:

         MD5:  6C:39:7D:A4:0E:55:59:B2:3F:D6:41:B1:12:50:DE:43

         SHA1: 5F:3B:8C:F2:F8:10:B3:7D:78:B4:CE:EC:19:19:C3:73:34:B9:C7:74

         Signature algorithm name: SHA256withRSA

         Version: 3


Extensions:


#1: ObjectId: 2.5.29.15 Criticality=true

KeyUsage [

  Key_CertSign

  Crl_Sign

]


#2: ObjectId: 2.5.29.19 Criticality=true

BasicConstraints:[

  CA:true

  PathLen:2147483647

]


#3: ObjectId: 2.5.29.14 Criticality=false

SubjectKeyIdentifier [

KeyIdentifier [

0000: 0A 85 A9 77 65 05 98 7C   40 81 F8 0F 97 2C 38 F1  ...we...@....,8.

0010: 0A EC 3C CF                                        ..<.

]

]


Trust this certificate? [no]:  ← 'yes'を入力

Certificate was added to keystore
```

- Oracle JDKの場合

```
$ sudo /usr/java/jdk1.6.0_16/bin/keytool -import -alias scrootca2 -keystore /usr/java/jdk1.6.0_16/jre/lib/security/cacerts -
file SCRoot2ca.cer
（内容はOpenJDKの場合と同じですので省略します）
```

証明書が導入されたことの確認

・トラストアンカー(cacerts)にSecurity Communication Root CA2 証明書 [SR3.0用]がインポートされていることを確認します。表示されたフィンガープリントがSecurity Communication Root CA2 証明書 [SR3.0用]と一致していることを確認してください。

- OpenJDK 1.6.0の場合

```
$ keytool -list -keystore /etc/alternatives/jre/lib/security/cacerts -alias scrootca2

Enter keystore password:   ← パスワードを変更していない場合はデフォルトのパスワード'changeit'を入力するか、

                             空のままEnterを入力します。

scrootca2, Dec 22, 2014, trustedCertEntry,

Certificate fingerprint (MD5): 6C:39:7D:A4:0E:55:59:B2:3F:D6:41:B1:12:50:DE:43

                                ↑ フィンガープリントを確認します
```

- OpenJDK 1.7.0の場合

```
$ keytool -list -keystore /etc/alternatives/jre/lib/security/cacerts -alias scrootca2

Enter keystore password:   ← パスワードを変更していない場合はデフォルトのパスワード'changeit'を入力するか、

                             空のままEnterを入力します。

scrootca2, Dec 24, 2014, trustedCertEntry,

Certificate fingerprint (SHA1): 5F:3B:8C:F2:F8:10:B3:7D:78:B4:CE:EC:19:19:C3:73:34:B9:C7:74

                                ↑ フィンガープリントを確認します
```

- Oracle JDKの場合

```
$ /usr/java/jdk1.6.0_16/bin/keytool -list -keystore /usr/java/jdk1.6.0_16/jre/lib/security/cacerts -alias scrootca2

Enter keystore password:   ← パスワードを変更していない場合はデフォルトのパスワード'changeit'を入力するか、

                             空のままEnterを入力します。

scrootca2, Jan 9, 2015, trustedCertEntry,

Certificate fingerprint (MD5): 6C:39:7D:A4:0E:55:59:B2:3F:D6:41:B1:12:50:DE:43

                                ↑ フィンガープリントを確認します
```