Orchestrating a brighter world

NEC

SecureWare/Credential Lifecycle Manager
コマンドリファレンス

2017 年 10 月

日本電気株式会社

はしがき

本書は、SecureWare/Credential Lifecycle Manager(以下、CLM と称します)が提供するコマンドについて説明したものです。

本書の構成は以下のとおりです。

章	タイトル	内容
1	提供コマンド一覧	CLM が提供するコマンドの概要説明
2	WebAPI 実行コマンド	WebAPI 実行コマンドについての説明

2017 年 10 月　第一版

目次

1　提供コマンド一覧

CLM で提供するコマンドは、下表の通りです。

コマンドの詳細は、2 章以降をご覧ください。

項番	コマンド	コマンド名	インストールディレクトリ	説明
1	WebAPI 実行コマンド	swcagent	/opt/nec/pf/swcagent/bin/	パラメータを指定することにより、各 Web API を実行する。

2 WebAPI 実行コマンド

WebAPI 実行コマンドは、CLM が提供する WebAPI を実行するコマンドです。

本コマンドを実行することで、CLM が提供する WebAPI の機能をコマンドラインから利用し、ID 鍵の管理を行うことができます。

また、本コマンドは、発行・取得・更新した証明書・共通鍵のファイル保存や証明書・共通鍵の改ざん検知など WebAPI 実行の他に付加機能を提供します。

前提

- コマンドは、root ユーザもしくは root グループに所属するユーザで実行してください。
- コマンド実行環境では、事前に以下の環境変数を設定しておく必要があります。環境変数を設定しない場合、コマンド実行時にエラーが発生し、コマンド実行に失敗しますのでご注意ください。

  LD_LIBRARY_PATH=/opt/nec/pf/swcagent/lib:$LD_LIBRARY_PATH

  SWSDKV50_LIBPATH=/opt/nec/pf/swcagent/lib

提供機能

本コマンドの提供している機能は以下の通りです。

- ID・パスワード発行

  ID・パスワード発行 API を実行し、エッジ－クラウド間の通信を安全に行うための ID・パスワード(事前共有キー)を発行します。パスワードの複雑さはサーバ側の設定に従います。
- ID・パスワード取得

  ID･パスワード取得 API を実行し、 ID･パスワード発行 API で発行した ID･パスワード(事前共有キー)を取得します。
- ID・パスワード照合

  ID・パスワード照合 API を実行し、WebAPI 実行コマンドが所有している ID・パスワード(事前共有キー)と、CLM に登録されている ID・パスワード(事前共有キー)が一致するかを照合(認証)します。
- ID・パスワード削除

  ID･パスワード削除 API を実行し、 ID･パスワード発行 API で発行した ID･パスワード(事前共有キー)を削除します。

- 証明書発行

  証明書発行 API を実行し、CA 証明書、公開鍵証明書を発行・取得します。また、発行・取得した証明書から証明書改ざん検知用のハッシュ値を生成し、ファイルに保存します。
- 証明書取得

証明書取得 API を実行し、CA 証明書、公開鍵証明書を取得します。また、取得した証明書から証明書改ざん検知用のハッシュ値を生成し、ファイルに保存します。

- 証明書更新

  証明書更新 API を実行し、公開鍵証明書を更新します。更新した証明書から証明書改ざん検知用のハッシュ値を生成し、ファイルに保存します。

- 証明書失効

  証明書失効 API を実行し、公開鍵証明書を失効します。

- 共通鍵発行

  共通鍵発行 API を実行し、共通鍵を発行します。また、発行した共通鍵から共通鍵改ざん検知用のハッシュ値を生成し、ファイルに保存します。

- 共通鍵取得

  証明書取得 API を実行し、CA 証明書、公開鍵証明書を取得します。また、取得した証明書から証明書改ざん検知用のハッシュ値を生成し、ファイルに保存します。

- 共通鍵更新

  共通鍵更新 API を実行し、共通鍵の更新します。更新した共通鍵から共通鍵改ざん検知用のハッシュ値を生成し、ファイルに保存します。

- 共通鍵削除

  共通鍵削除 API を実行し、共通鍵を削除します。

実行形式

実行時に以下のオプション、および、オプション値を指定します。指定できるオプションは、機能によって異なります。

また、証明書発行・取得・更新・失効と、共通鍵発行・取得・更新・削除コマンドを実行する場合、事前にID・パスワード照合コマンドを実行し、CLM と ID・パスワード(事前共有キー)による認証を行っておく必要があります。

- ID・パスワード発行

> swcagent idcreate	-host <hostname> [-port <port >] [-p-host <hostname >] [-p-port <port >] [-timeout <timeout>] -ext-data <extension data>

オプション	必須/任意	説明
-host	必須	CLM のホスト名または IP アドレスを指定します。
-port	必須	CLM の待ち受けポート番号を指定します。既定値は「8443」です。
-p-host	任意	プロキシサーバの IP アドレスを指定します。
-p-port	任意	プロキシサーバのポート番号を指定します。
-timeout	任意	タイムアウト時間を指定します。単位は秒です。既定値は 60 秒です。指定可能な値は 1～0x7FFFFFFF です。
-extdata	必須	拡張情報を指定します。 key=value の形式で指定してください。 value は、クォートで囲む必要があります。また、複数指定する場合はカンマで区切る必要があります。 指定する拡張情報は、後述の「拡張情報」を参照してください。

拡張情報

ID・パスワード発行機能では、-extdata に以下の項目を指定することができます。

項目	必須/任意	説明
tenantid	必須	テナント ID を指定します。
alias	必須	ID・パスワードに付与する Alias 名を指定します、

実行例

```
#swcagen idcreate -host "192.168.0.1" -port "8443" -ext-data tenantid="TenantA",alias="
iotgatewaytest"
```

● ID・パスワード取得

> swcagent idget	-host <hostname> [-port <port >] [-p-host <hostname >] [-p-port <port >] [-timeout <timeout>] -ext-data <extension data>

オプション	必須/任意	説明
-host	必須	CLM のホスト名または IP アドレスを指定します。
-port	必須	CLM の待ち受けポート番号を指定します。既定値は「8443」です。
-p-host	任意	プロキシサーバの IP アドレスを指定します。
-p-port	任意	プロキシサーバのポート番号を指定します。
-timeout	任意	タイムアウト時間を指定します。単位は秒です。既定値は 60 秒です。指定可能な値は 1～0x7FFFFFFF です。
-extdata	必須	拡張情報を指定します。 key=value の形式で指定してください。 value は、クォートで囲む必要があります。また、複数指定する場合はカンマで区切る必要があります。 指定する拡張情報は、後述の「拡張情報」を参照してください。

拡張情報

ID・パスワード取得機能では、-extdata に以下の項目を指定することができます

項目	必須/任意	説明
keyid	必須	ID・パスワード発行 API で発行した ID を指定します。

実行例

```
# swcagent idget -host "192.168.0.1" -port "8443" -ext-data keyid="0000000019"
```

● ID・パスワード照合

> swcagent idverify	-host <hostname> [-port <port >] [-p-host <hostname >] [-p-port <port >] [-timeout <timeout>] -ext-data <extension data>

オプション	必須/任意	説明
-host	必須	CLM のホスト名または IP アドレスを指定します。
-port	必須	CLM の待ち受けポート番号を指定します。既定値は「8443」です。
-p-host	任意	プロキシサーバの IP アドレスを指定します。
-p-port	任意	プロキシサーバのポート番号を指定します。
-timeout	任意	タイムアウト時間を指定します。単位は秒です。既定値は 60 秒です。指定可能な値は 1～0x7FFFFFFF です。
-extdata	必須	拡張情報を指定します。 key=value の形式で指定してください。 value は、クォートで囲む必要があります。また、複数指定する場合はカンマで区切る必要があります。 指定する拡張情報は、後述の「拡張情報」を参照してください。

拡張情報

ID・パスワード照合機能では、-extdata に以下の項目を指定することができます。

項目	必須/任意	説明
keyid	必須	ID・パスワード発行 API で発行した ID を指定します。
key	必須	ID・パスワード発行 API で発行した ID のパスワードを指定します。

実行例

```
# swcagent idverify -host "192.168.0.1" -port "8443" -ext-data keyid="0000000001",key="UM9Po1uJ"
```

● ID・パスワード削除

> swcagent iddelete	-host <hostname> [-port <port >] [-p-host <hostname >] [-p-port <port >] [-timeout <timeout>] -ext-data <extension data>

オプション	必須/任意	説明
-host	必須	CLM のホスト名または IP アドレスを指定します。
-port	必須	CLM の待ち受けポート番号を指定します。既定値は「8443」です。
-p-host	任意	プロキシサーバの IP アドレスを指定します。
-p-port	任意	プロキシサーバのポート番号を指定します。
-timeout	任意	タイムアウト時間を指定します。単位は秒です。既定値は 60 秒です。指定可能な値は 1～0x7FFFFFFF です。
-extdata	必須	拡張情報を指定します。 key=value の形式で指定してください。 value は、クォートで囲む必要があります。また、複数指定する場合はカンマで区切る必要があります。 指定する拡張情報は、後述の「拡張情報」を参照してください。

拡張情報

ID・パスワード削除機能では、-extdata に以下の項目を指定することができます。

項目	必須/任意	説明
keyid	必須	ID・パスワード発行 API で発行した ID を指定します。
key	必須	ID・パスワード発行 API で発行した ID のパスワードを指定します。

実行例

```
# swcagent iddelete -host "192.168.0.1" -port "8443" -ext-data keyid="0000000001",key
="UM9Po1uJ"
```

● 証明書発行

> swcagent certcreate	-host <hostname> [-port <port >] [-p-host <hostname >] [-p-port <port >] [-timeout <timeout>] -ext-data <extension data> -mode <mode> [-cacerttype <type>] [-certtype <type>] [-passphrase <passphrase>] -outpath <output directory> [-cacertname <filename>] [-certname <filename>] [-certkeyname <filename>]

オプション	必須/任意	説明
-host	必須	CLM のホスト名または IP アドレスを指定します。
-port	必須	CLM の待ち受けポート番号を指定します。既定値は「8443」です。
-p-host	任意	プロキシサーバの IP アドレスを指定します。
-p-port	任意	プロキシサーバのポート番号を指定します。
-timeout	任意	タイムアウト時間を指定します。単位は秒です。既定値は 60 秒です。指定可能な値は 1～0x7FFFFFFF です。
-extdata	必須	拡張情報を指定します。 key=value の形式で指定してください。 value は、クォートで囲む必要があります。また、複数指定する場合はカンマで区切る必要があります。 指定する拡張情報は、後述の「拡張情報」を参照してください。
-mode	必須	取得する証明書の種別を指定します。 以下のいずれかを指定してください。 caonly: CA 証明書を取得する client: クライアント証明書を取得する server: サーバ証明書を取得する
-cacerttype	任意	CA 証明書の種別を指定します。

		以下のいずれかを指定してください。 pem: pem 形式の証明書 (本項目省略時の既定値) der：der 形式の証明書
-certtype	任意	サーバ証明書、クライアント証明書の種別を指定します。 以下のいずれかを指定してください。 pem: pem 形式の証明書 (本項目省略時の既定値) der：der 形式の証明書 p12：PKCS#12 形式の証明書(chain なし)
-passphrase	任意 ※certtype が p12 の場合は、必須です。	サーバ証明書、クライアント証明書の秘密鍵のパスフレーズを指定します。 クライアント証明書の種別に「p12」が指定された場合、PKCS#12 形式に変換する際のパスフレーズにも利用します。
-outpath	必須	証明書出力ディレクトリを指定します。
-cacertname	任意	取得した CA 証明書を保存するファイル名を指定します。拡張子は不要です。既定値は「cacert」です。
-certname	任意	取得したサーバ証明書、クライアント証明書を保存するファイル名を指定します。 拡張子は不要です。既定値は以下の通りです。 mode が「client」である場合 ：「clcert」 mode が「server」である場合:「svcert」
-certkeyname	任意	取得したクライアント証明書の秘密鍵を保存するファイル名を指定します。 拡張子は不要です。既定値は以下の通りです。 mode が「client」である場合 ：「clkey」 mode が「server」である場合:「svkey」

拡張情報

証明書発行機能では、-extdata に以下の項目を指定することができます。

項目	必須/任意	説明
devicename	必須	デバイス名。証明書発行または共通鍵発行時に指定した名称を指定します。
catype	必須	証明書の認証局名称を指定します。 ・CA 証明書を取得したい場合、「CA」を指定してください。 ・公開鍵証明書を発行・取得・更新・失効したい場合、「ca1」を指定してください。
certsubject	必須	証明書のサブジェクトを指定します。

		「/C=国別コード/ST=都道府県名/L=市区町村名/O=部門名/OU=組織名/CN=コモンネーム」の形式で指定してください。「/CN=コモンネーム」以外は省略可能です。

実行例

[CA 証明書発行]

```
# swcagent certcreate -host "192.168.0.1" -port "8443" -mode "caonly" -cacerttype "pem" -outpath "/home/iotgateway" -cacertname "iotgatewaytest_ca" -ext-data devicename="iotgatewaytest",catype="CA"
```

[クライアント証明書(PEM)]

```
# swcagent certcreate -host "192.168.0.1" -port "8443" -mode "client" -cacerttype "pem" -certtype "pem” -outpath "/home/iotgateway" -cacertname "iotgatewaytest_ca" -certname "iotgatewaytest_clcert" -certkeyname "iotgatewaytest_clientkey" -ext-data certsubject="/C=JP/O=NEC/CN=iotgatewaytest",devicename="iotgatewaytest",catype="ca1"
```

[クライアント証明書発行(PKCS#12)]

```
# swcagent certcreate -host "192.168.0.1" -port "8443" -mode “client” -certtype “p12” -outpath “/home/iotgateway” -certname “iotgatewaytest_clcert” -passphrase "password" -ext-data certsubject="/C=JP/O=NEC/CN=iotgatewaytest",devicename="iotgatewaytest",catype="ca1"
```

[サーバ証明書発行(DER)]

```
# swcagent certcreate -host "192.168.0.1" -port "8443" -mode “server” -cacerttype “der” –certtype “der” -outpath “/home/iotgateway” –cacertname “iotgatewaytest_ca” -certname “iotgatewaytest_svcert” -certkeyname “iotgatewaytest_svkey” -ext-data certsubject=“/C=JP/O=NEC/CN=iotgatewaytest",devicename="iotgatewaytest",catype="ca1"
```

● 証明書取得

> swcagent certget	-host <hostname> [-port <port >] [-p-host <hostname >] [-p-port <port >] [-timeout <timeout>] -ext-data <extension data> [-certtype <type>] [-passphrase <passphrase>] -outpath <output directory> [-certname <filename>] [-certkeyname <filename>]

オプション	必須/任意	説明
-host	必須	CLM のホスト名または IP アドレスを指定します。
-port	任意	CLM の待ち受けポート番号を指定します。既定値は「8443」です。
-p-host	任意	プロキシサーバの IP アドレスを指定します。
-p-port	任意	プロキシサーバのポート番号を指定します。
-timeout	任意	タイムアウト時間を指定します。単位は秒です。既定値は 60 秒です。指定可能な値は 1～0x7FFFFFFF です。
-extdata	必須	拡張情報を指定します。 key=value の形式で指定してください。 value は、クォートで囲む必要があります。また、複数指定する場合はカンマで区切る必要があります。 指定する拡張情報は、後述の「拡張情報」を参照してください。
-certtype	任意	取得対象証明書の種別を指定します。 pem: pem 形式の証明書 (本項目省略時の既定値) der：der 形式の証明書 p12：PKCS#12 形式の証明書(chain なし)
-passphrase	任意 ※certtype が p12 の場合、必須です。	取得対象証明書の秘密鍵のパスフレーズを指定します。 取得対象証明書の種別に「p12」が指定された場合、PKCS#12 形式に変換する際のパスフレーズにも利用します。
-outpath	必須	取得対象証明書出力ディレクトリを指定します。

-certname	任意	取得対象証明書を保存するファイル名を指定します。拡張子は不要です。既定値は「cert」です。
-certkeyname	任意	取得対象証明書の秘密鍵を保存するファイル名を指定します。拡張子は不要です。既定値は「certkey」です。

拡張情報

証明書取得機能では、-extdata に以下の項目を指定することができます。

項目	必須/任意	説明
devicename	必須	デバイス名。証明書発行時に指定した名称を指定します。
catype	必須	証明書の認証局名称を指定します。 ・CA 証明書を取得したい場合、「CA」を指定してください。 ・公開鍵証明書を発行・取得・更新・失効したい場合、「ca1」を指定してください。
keynumber	必須 ※1	対象証明書の鍵番号。 ※1：certserial を指定しない場合は任意です
certserial	必須 ※2	対象証明書のシリアル番号。 ※2：keynumber を指定しない場合は任意です。

実行例

[クライアント証明書取得 (PEM 形式、シリアル番号指定)]

```
# swcagent certget -host “192.168.0.1” -port “8443” -certtype “pem” -outpath “/home/iotgateway” -certname “iotgatewaytest_clcert” -certkeyname “iotgatewaytest_clientkey” -ext-data devicename=“iotgatewaytest”,certserial=“CEC798689CF69040”,catype=“ca1”
```

[クライアント証明書取得 (PKCS#12 形式、鍵番号指定)]

```
# swcagent certget -host “192.168.0.1” -port “8443” -certtype “p12” -outpath “/home/iotgateway” -certname “iotgatewaytest_clcert” -passphrase “password” -ext-data devicename=“iotgatewaytest”,keynumber=“39”,catype=“ca1“
```

[サーバ証明書取得 (DER 形式、シリアル番号指定)]

```
# swcagent certget -host "192.168.0.1" -port "8443" -certtype “der” -outpath “/home/iotgateway” -certname “iotgatewaytest_svcert” -certkeyname “iotgatewaytest_svkey" -ext-data devicename="iotgatewaytest",certserial="DEBAB6D4A8257CA5",catype="ca1"
```

● 証明書更新

> swcagent certupdate	-host <hostname> [-port <port >] [-p-host <hostname >] [-p-port <port >] [-timeout <timeout>] -ext-data <extension data> [-certtype <type>] [-passphrase <passphrase>] -outpath <output directory> [-certname <filename>]

オプション	必須/任意	説明
-host	必須	CLM のホスト名または IP アドレスを指定します。
-port	任意	CLM の待ち受けポート番号を指定します。既定値は「8443」です。
-p-host	任意	プロキシサーバの IP アドレスを指定します。
-p-port	任意	プロキシサーバのポート番号を指定します。
-timeout	任意	タイムアウト時間を指定します。単位は秒です。既定値は 60 秒です。指定可能な値は 1～0x7FFFFFFF です。
-extdata	必須	拡張情報を指定します。 key=value の形式で指定してください。 value は、クォートで囲む必要があります。また、複数指定する場合はカンマで区切る必要があります。 指定する拡張情報は、後述の「拡張情報」を参照してください。
-certtype	任意	取得対象証明書の種別を指定します。 pem: pem 形式の証明書（本項目省略時の既定値) der：der 形式の証明書 p12：PKCS#12 形式の証明書(chain なし)
-passphrase	任意 ※certtype が p12 の場合、必須です。	更新後の証明書の秘密鍵のパスフレーズを指定します。 取得対象証明書の種別に「p12」が指定された場合、PKCS#12 形式に変換する際のパスフレーズにも利用します。
-outpath	必須	更新対象証明書が保存されているディレクトリを指定します。
-certname	任意	更新対象証明書ファイル名を指定します。拡張子は不要です。

		既定値は「cert」です。
-certkeyname	任意	取得対象証明書の秘密鍵を保存するファイル名を指定します。拡張子は不要です。既定値は「certkey」です。

拡張情報

証明書更新機能では、-extdata に以下の項目を指定することができます。

項目	必須/任意	説明
devicename	必須	デバイス名。証明書発行時に指定した名称を指定します。
catype	必須	証明書の認証局名称を指定します。 ・CA 証明書を取得したい場合、「CA」を指定してください。 ・公開鍵証明書を発行・取得・更新・失効したい場合、「ca1」を指定してください。
keynumber	必須 ※1	対象証明書の鍵番号を指定します。 ※1：certserial を指定しない場合は任意です
certserial	必須 ※2	対象証明書のシリアル番号を指定します。 ※2：keynumber を指定しない場合は任意です。
newkeynumber	任意	更新後の証明書の鍵番号を指定します。 既存の証明書を更新後証明書としたい場合に指定してください。
newcertserial	任意	更新後の証明書のシリアル番号を指定します。 既存の証明書を更新後証明書としたい場合に指定してください。
newcerttype	任意	更新後の証明書の種別を指定します。 以下のいずれかを指定してください。 pem: pem 形式の証明書 (本項目省略時の既定値) der：der 形式の証明書 p12：PKCS#12 形式の証明書(chain なし)
newpath	任意	更新後の証明書出力ディレクトリを指定します。 -outpath に指定したディレクトリと異なるディレクトリに出力する場合に指定してください。
newfilename	任意	更新した証明書を保存するファイル名を指定します。 拡張子は不要です。 -certname に指定したファイル名と異なるファイル名を付与する場合に指定してください。

実行例

[証明書更新（PEM 形式、シリアル番号指定、更新後証明書を新規発行)]

```
# swcagent certupdate -host “192.168.0.1” -port “8443” -certtype “pem” -outpath “/home/iotgateway“ -certname ”iotgatewaytest_clcert” -ext-data devicename=“iotgatewaytest“,catype=”ca1”,certserial=“CEC798689CF69040”,newpath=“/home/iotgateway”,newfilename=“iotgatewaytest_clcert _new“,newcerttype=”pem”
```

[証明書更新 (PEM 形式、鍵番号指定、既存の証明書で更新)]

```
# swcagent certupdate -host “192.168.0.1” -port “8443” -certtype “pem” -outpath “/home/iotgateway“ -certname ”iotgatewaytest_clcert” -ext-data devicename=“iotgatewaytest”,catype=“ca1“,keynumber=”64”,newkeynumber=“25”,newpath=“/home/iotgateway“,newfilename="iotgatewaytest_clcert_new",newcerttype="pem“
```

● 証明書失効

> swcagent certrevoke	-host <hostname> [-port <port >] [-p-host <hostname >] [-p-port <port >] [-timeout <timeout>] -ext-data <extension data> [-certtype <type>] [-certname <filename>]

オプション	必須/任意	説明
-host	必須	CLM のホスト名または IP アドレスを指定します。
-port	任意	CLM の待ち受けポート番号を指定します。既定値は「8443」です。
-p-host	任意	プロキシサーバの IP アドレスを指定します。
-p-port	任意	プロキシサーバのポート番号を指定します。
-timeout	任意	タイムアウト時間を指定します。単位は秒です。既定値は 60 秒です。指定可能な値は 1～0x7FFFFFFF です。
-extdata	必須	拡張情報を指定します。 key=value の形式で指定してください。 value は、クォートで囲む必要があります。また、複数指定する場合はカンマで区切る必要があります。 指定する拡張情報は、後述の「拡張情報」を参照してください。
-certtype	任意	失効対象証明書の種別を指定します。 pem: pem 形式の証明書 (本項目省略時の既定値) der：der 形式の証明書 p12：PKCS#12 形式の証明書(chain なし)
-certname	必須	更新対象証明書ファイル名を指定します。拡張子は不要です。

拡張情報

証明書失効機能では、-extdata に以下の項目を指定することができます。

項目	必須/任意	説明
devicename	必須	デバイス名。証明書発行時に指定した名称を指定します。
catype	必須	証明書の認証局名称を指定します。 ・CA 証明書を取得したい場合、「CA」を指定してください。

		・公開鍵証明書を発行・取得・更新・失効したい場合、「ca1」を指定してください。
keynumber	必須 ※1	対象証明書の鍵番号を指定します。 ※1：certserial を指定しない場合は任意です
certserial	必須 ※2	対象証明書のシリアル番号を指定します。 ※2：keynumber を指定しない場合は任意です。
inpath	必須	失効対象証明書が保存されているディレクトリ。

実行例

[証明書失効 (鍵番号指定)]

```
# swcagent certrevoke -host “192.168.0.1” -port “8443” -certtype “pem“ -certname=”iot
  gatewaytest_clcert“ -ext-data devicename=”iotgatewaytest”,catype=“ca1”,keynumber=“6
  5”,inpath=“/home/iotgateway“
```

[証明書失効 (シリアル番号指定)]

```
# swcagent certrevoke -host “192.168.0.1” -port “8443” -certtype “pem“ -certname=”iot
  gatewaytest_clcert“ -ext-data devicename=”iotgatewaytest”,catype=“ca1”,certserial=“C
  EC798689CF69040”,inpath=“/home/iotgateway“
```

● 共通鍵発行

> swcagent cmkeycreate	-host <hostname> [-port <port >] [-p-host <hostname >] [-p-port <port >] [-timeout <timeout>] -ext-data <extension data> -outpath <output directory>

オプション	必須/任意	説明
-host	必須	CLM のホスト名または IP アドレスを指定します。
-port	任意	CLM の待ち受けポート番号を指定します。既定値は「8443」です。
-p-host	任意	プロキシサーバの IP アドレスを指定します。
-p-port	任意	プロキシサーバのポート番号を指定します。
-timeout	任意	タイムアウト時間を指定します。単位は秒です。既定値は 60 秒です。指定可能な値は 1～0x7FFFFFFF です。
-extdata	必須	拡張情報を指定します。 key=value の形式で指定してください。 value は、クォートで囲む必要があります。また、複数指定する場合はカンマで区切る必要があります。 指定する拡張情報は、後述の「拡張情報」を参照してください。
-outpath	必須	共通鍵を出力するディレクトリを指定します。

拡張情報

共通鍵発行機能では、-extdata に以下の項目を指定することができます。

項目	必須/任意	説明
devicename	必須	デバイス名。証明書発行または共通鍵発行時に指定した名称を指定します。
alias	必須	共通鍵の Alias 名を指定します。
cmkeyname	任意	発行・取得した共通鍵を保存するファイル名を指定します。 省略時の既定値は「cmkey」です。
keylength	任意	発行する共通鍵の長さ(単位: bit)を指定します。 省略時の既定値は 128(bit)です。

keytype	必須	鍵を利用する暗号化方式を指定します。 以下のいずれかを指定してください。 AES　：AES で利用する TWINE：TWINE で利用する

実行例

[共通鍵発行(AES 用 鍵長 128bit)]

```
# swcagent cmkeycreate -host “192.168.0.1” -port “8443” -outpath “/home/iotgateway“
  -ext-data devicename=”iotgatewaytest”,alias=“iotgatewaytest aes key”,keytype=“AES“,c
  mkeyname="iotgatewaytest_cmkey_aes",keylength="128“
```

● 共通鍵取得

> swcagent cmkeyget	-host <hostname> [-port <port >] [-p-host <hostname >] [-p-port <port >] [-timeout <timeout>] -ext-data <extension data> -outpath <output directory>

オプション	必須/任意	説明
-host	必須	CLM のホスト名または IP アドレスを指定します。
-port	任意	CLM の待ち受けポート番号を指定します。既定値は「8443」です。
-p-host	任意	プロキシサーバの IP アドレスを指定します。
-p-port	任意	プロキシサーバのポート番号を指定します。
-timeout	任意	タイムアウト時間を指定します。単位は秒です。既定値は 60 秒です。指定可能な値は 1～0x7FFFFFFF です。
-extdata	必須	拡張情報を指定します。 key=value の形式で指定してください。 value は、クォートで囲む必要があります。また、複数指定する場合はカンマで区切る必要があります。 指定する拡張情報は、後述の「拡張情報」を参照してください。
-outpath	必須	共通鍵を出力するディレクトリを指定します。

拡張情報

共通鍵取得機能では、-extdata に以下の項目を指定することができます。

項目	必須/任意	説明
devicename	必須	デバイス名。共通鍵発行時に指定した名称を指定します。
cmkeyname	任意	発行・取得した共通鍵を保存するファイル名を指定します。 省略時の既定値は「cmkey」です。
keylength	任意	発行する共通鍵の長さ(単位: bit)を指定します。 省略時の既定値は 128(bit)です。

[共通鍵発行(AES 用 鍵長 128bit)]

```
# swcagent cmkeycreate -host “192.168.0.1” -port “8443” -outpath “/home/iotgateway“
  -ext-data devicename=”iotgatewaytest”,alias=“iotgatewaytest aes key”,keytype=“AES“,c
  mkeyname="iotgatewaytest_cmkey_aes",keylength="128“
```

● 共通鍵更新

> swcagent cmkeyupdate	-host <hostname> [-port <port >] [-p-host <hostname >] [-p-port <port >] [-timeout <timeout>] -ext-data <extension data> -outpath <output directory>

オプション	必須/任意	説明
-host	必須	CLM のホスト名または IP アドレスを指定します。
-port	任意	CLM の待ち受けポート番号を指定します。既定値は「8443」です。
-p-host	任意	プロキシサーバの IP アドレスを指定します。
-p-port	任意	プロキシサーバのポート番号を指定します。
-timeout	任意	タイムアウト時間を指定します。単位は秒です。既定値は 60 秒です。指定可能な値は 1～0x7FFFFFFF です。
-extdata	必須	拡張情報を指定します。 key=value の形式で指定してください。 value は、クォートで囲む必要があります。また、複数指定する場合はカンマで区切る必要があります。 指定する拡張情報は、後述の「拡張情報」を参照してください。
-outpath	必須	共通鍵を出力するディレクトリを指定します。

拡張情報

共通鍵更新機能では、-extdata に以下の項目を指定することができます。

項目	必須/任意	説明
devicename	必須	デバイス名。共通鍵発行時に指定した名称を指定します。
alias	任意	共通鍵の Alias を指定します。
cmkeyname	任意	発行・取得した共通鍵を保存するファイル名を指定します。 省略時の既定値は「cmkey」です。
newpath	任意	更新後の共通鍵出力ディレクトリを指定します。 -outpath に指定したディレクトリと異なるディレクトリに出力する場合に指定してください。

newfilename	任意	発行する共通鍵の長さ(単位: bit)を指定します。 省略時の既定値は 128(bit)です。

実行例

[共通鍵更新（AES 用 鍵長 128bit、更新後共通鍵を新規発行)]

```
# swcagent cmkeyupdate -host “192.168.0.1” -port “8443” -outpath “/home/iotgateway“
    -ext-data devicename=”iotgatewaytest”,keynumber=“66”,
    cmkeyname=“iotgatewaytest_cmkey_aes“,
    keylength=“128”,newpath=”/home/iotgateway”,
    newfilename=“iotgatewaytest_cmkey_aes_new”
```

[共通鍵更新（AES 用 鍵長 128bit、既存の共通鍵で更新)]

```
# swcagent cmkeyupdate -host “192.168.0.1” -port “8443” -outpath “/home/iotgateway
    “ -ext-data devicename=”iotgatewaytest”,keynumber=“66”,newkeynumber=“41“,cmkey
    name=”iotgatewaytest_cmkey_aes”,keylength=“128”,newpath=“/home/iotgateway,newfi
    lename="iotgatewaytest_cmkey_aes_new"
```

● 共通鍵削除

> swcagent cmkeydelete	-host <hostname> [-port <port >] [-p-host <hostname >] [-p-port <port >] [-timeout <timeout>] -ext-data <extension data>

オプション	必須/任意	説明
-host	必須	CLM のホスト名または IP アドレスを指定します。
-port	任意	CLM の待ち受けポート番号を指定します。既定値は「8443」です。
-p-host	任意	プロキシサーバの IP アドレスを指定します。
-p-port	任意	プロキシサーバのポート番号を指定します。
-timeout	任意	タイムアウト時間を指定します。単位は秒です。既定値は 60 秒です。指定可能な値は 1～0x7FFFFFFF です。
-extdata	必須	拡張情報を指定します。 key=value の形式で指定してください。 value は、クォートで囲む必要があります。また、複数指定する場合はカンマで区切る必要があります。 指定する拡張情報は、後述の「拡張情報」を参照してください。

拡張情報

共通鍵更新機能では、-extdata に以下の項目を指定することができます。

項目	必須/任意	説明
devicename	必須	デバイス名。共通鍵発行時に指定した名称を指定します。
cmkeyname	必須	発行・取得した共通鍵を保存するファイル名を指定します。 省略時の既定値は「cmkey」です。
inpath	必須	削除対象共通鍵が保存されているディレクトリを指定します。

実行例

[共通鍵削除]

```
# swcagent cmkeydelete -host “192.168.0.1” -port “8443” -ext-data devicename=“iotgat
  ewaytest”, keynumber="66",inpath="/home/iotgateway",cmkeyname="iotgatewaytest_c
```

```
mkey_aes"
```

実行結果

実行結果は、WebAPI 実行コマンドの終了コードと CLM からのレスポンスで確認します。

WebAPI 実行コマンド 終了コード

終了コードについては、3 章をご覧ください。また、必要に応じ、別紙「WebAPI リファレンス」のエラーコード一覧をご覧ください。

CLM からのレスポンス

WebAPI 実行コマンドを実行すると、CLM からのレスポンスをファイルに出力し、以下に保存します。

```
/tmp/swclm_result.json
```

CLM からのレスポンスは JSON 形式です。レスポンスの詳細は、別紙「WebAPI リファレンス」記載の各 API レスポンスパラメータをご覧ください。

「証明書発行・取得・更新」コマンドで発行・取得・更新した証明書について

CLM で発行した証明書は、WebAPI 実行コマンド 実行環境上に保存します。保存先は、次の通りです。

- 証明書発行、証明書取得時

  パラメータ「-outpath」に指定したディレクトリです。
- 証明書更新時

  パラメータ「-outpath」に指定したディレクトリです。ただし、拡張情報「newpath」を指定している場合は、拡張情報「newpath」に指定したディレクトリに保存します。

保存する証明書のファイル名は、次の通りです。

- CA 証明書

  パラメータ「-cacertname」に指定した値がファイル名となります。
- 公開鍵証明書

  パラメータ「-certname」に指定した値がファイル名となります。

  ただし、証明書更新時、拡張情報「newfilename」を指定している場合は、拡張情報「newfilename」に指定した値がファイル名となります。
- 公開鍵証明書の秘密鍵

  パラメータ「-certkeyname」にした値がファイル名となります。

保存する証明書の拡張子は、次の通りです。

- CA 証明書

パラメータ「-cacerttype」に指定した値に従い付与されます。

- 公開鍵証明書

  パラメータ「-certtype」に指定した値に従い付与されます。

  ただし、証明書更新時、拡張情報「newcerttype」を指定している場合は、拡張情報「 newcerttype 」に指定した値に従い付与されます。

- 公開鍵証明書の秘密鍵

  拡張子は「.pem」固定です

「共通鍵発行・取得・更新」機能で発行・取得・更新した共通鍵について

CLM で発行した共通鍵は、WebAPI 実行コマンド 実行環境上に保存します。保存先は、次の通りです。

- 共通鍵発行、共通鍵取得時

  パラメータ「-outpath」に指定したディレクトリです。

- 共通鍵更新時

  パラメータ「-outpath」に指定したディレクトリです。

  ただし、拡張情報「newpath」を指定している場合は、拡張情報「newpath」に指定したディレクトリに保存します。

保存する共通鍵のファイル名は、次の通りです。

- 共通鍵発行、共通鍵取得時

  拡張情報「cmkeyname 」に指定した値がファイル名となります。

- 共通鍵更新

  拡張情報「cmkeyname 」に指定した値がファイル名となります。

  ただし、拡張情報「newfilename」を指定している場合は、拡張情報「newfilename」に指定した値がファイル名となります。

ログ出力

WebAPI 実行コマンドは、以下にログを出力します。

- 標準出力・エラー出力
- /var/log/swcagent.log

3 終了コード一覧

3.1 WebAPI 実行コマンド 終了コード一覧

終了コードは以下の通りです。

エラーコード	説明
0	正常終了
1	コマンドのパラメータが不正
2	ファイル読み込み失敗
3	conf ファイル内パラメータ不正
4	必須パラメータが設定されていない
5	コマンドの書式が不正
6	既に実行中
7	メモリ確保エラー
8	ファイル書き込み失敗
11	取得した証明書の展開に失敗
12	取得した証明書の復号に失敗
13	証明書ストアへの取得した証明書保存に失敗
14	証明書が既に存在
15	証明書が改ざんされている
101	CLM への接続に失敗(Connection timeout)
102	CLM への接続に失敗(Connection refused)
103	暗号鍵の生成に失敗
104	認証プロトコルが不一致
105	プロトコルメッセージの送信に失敗
106	プロトコルメッセージの受信に失敗
107	プロトコルメッセージの暗号化に失敗
108	プロトコルメッセージの復号に失敗
201	CLM から返却された終了コード(要求応答不正(HTTP エラー))
255	内部エラー