セキュリティ

主要なセキュリティ概念

製品のセキュリティの最適化を目指す管理者は、次のセキュリティ概念をよく理解しておく必要があります。

HTTPS

Hypertext Transfer Protocol Secure（HTTPS）では、チャネルを介して送信されるデータの暗号化に、セキュア ソケット レイヤ（SSL） またはその後続の標準規格である Transport Layer Security（TLS）が使用されます。SSL で複数の脆弱性が見つかったため、 では現在 TLS のみがサポートされています。

（注） TLS は大まかに SSL と呼ばれることが多いため、本ガイドでもこの表記に従います。

SSL は、プライバシー、認証、およびデータ整合性を組み合わせることで、クライアントとサーバーの間のデータ転送を保護します。これらのセキュリティ メカニズムを有効にするために、SSL は証明書、秘密キー/公開キー交換ペア、および Diffie-Hellman 鍵共有パラメータを使用します。

SSL 証明書

SSL 証明書と秘密キー/公開キー ペアは、ユーザー認証および通信パートナーの ID 検証に使われるデジタル ID の一種です。VeriSign や Thawte などの認証局（CA）は、エンティティ（サーバーまたはクライアント）を識別するための証明書を発行します。クライアントまたはサーバー証明書には、発行認証局の名前とデジタル署名、シリアル番号、証明書が発行されたクライアントまたはサーバーの名前、公開キー、および証明書の有効期限が含まれます。CA は、1 つ以上の署名証明書を使用して SSL 証明書を作成します。各署名証明書には、CA 署名の作

成に使用される照合秘密キーがあります。CAは署名付き証明書（公開キーが埋め込まている）を簡単に入手できるようにしているため、誰でもその証明書を使用して、SSL証明書が実際に特定の CA によって署名されたことを確認できます。

一般に、証明書の設定には次の手順が含まれます。

1. サーバーの ID 証明書を生成する。
2. サーバーに ID 証明書をインストールする。
3. 対応するルート証明書をクライアントまたはブラウザにインストールする。

実行する必要がある具体的なタスクは、ご利用の環境によって異なります。

1 方向 SSL 認証

これは、クライアントが適切なサーバー（中間サーバーではなく）に接続していることを保証する必要がある場合に使用される認証方法で、オンライン バンキングの Web サイトなどのパブリック リソースに適しています。認証は、クライアントがサーバー上のリソースへのアクセスを要求したときに開始されます。リソースが存在するサーバーは、その ID を証明するために、サーバー証明書（別名 SSL 証明書）をクライアントに送信します。クライアントは受信したサーバー証明書を、クライアントまたはブラウザにインストールする必要がある別の信頼できるオブジェクト（サーバールート証明書）と照合して検証します。サーバーの検証後、暗号化された（つまりセキュアな）通信チャネルが確立されます。ここで、 サーバーは HTML フォームへの有効なユーザー名とパスワードの入力を求めます。SSL 接続が確立された後にユーザー クレデンシャルを入力すると、未認証の第三者による傍受を防ぐことができます。最終的に、ユーザー名とパスワードが受け入れられた後、サーバー上に存在するリソースへのアクセスが許可されます。

（注） クライアントは複数のサーバーとやり取りするために、複数のサーバー証明書を格納する必要がある場合があります。

クライアントにルート証明書をインストールする必要があるかどうかを判断するには、ブラウザの URL フィールドでロック アイコンを探します。通常このアイコンが表示される場合は、必要なルート証明書がすでにインストール済みであることを示します。多くの場合、これはより大きいいずれかの認証局（CA）によって署名されたサーバー証明書に該当します。一般的なブラウザではこれらの CA からのルート証明書が含まれているからです。

クライアントがサーバー証明書に署名した CA を認識しない場合は、接続がセキュリティで保護されていないことを意味します。これは必ずしも大きな問題ではなく、接続するサーバーの ID が検証されていないことを示しているだけです。この時点で、次の 2 つの操作のいずれかを実行できます。1 つは必要なルート証明書をクライアントまたはブラウザにインストールできます。ブラウザの URL フィールドにロック アイコンが表示された場合は、証明書が正常にインストールされたことを意味します。もう 1 つは、クライアントに自己署名証明書をインストールできることです。信頼できる CA によって署名されたルート証明書とは異なり、自己署名証明書は作成者である個人またはエンティティによって署名されます。自己署名証明書を使用して暗号化チャネルを作成できますが、接続するサーバーの ID が検証されていないため、固有のリスクが伴うことを理解しておいてください。

証明書のインストール

このセクションには、Cisco WAE サーバーへのセキュリティ証明書のインストール、Cisco WAE の調整されたメンテナンス、および Cisco WAE Liveに関する情報が含まれています。

Cisco WAE サーバーの証明書のインストール

Cisco WAE にはデフォルトの証明書が付属しています。この証明書は「信頼された CA」からのものではないため、ブラウザには保護されていない接続の警告が表示されます。これは予期されている動作です。この警告は、適切な認証局（CA）発行の証明書を適用することで削除できます。

ステップ 1 プライベートサーバーキーを作成し、安全な場所に保存します。次に例を示します。

```
# openssl genrsa -out server.key 2048
```

ステップ 2 証明書署名要求（CSR）を作成します。CAは、CSR を使用して、Web サイトを安全であると識別する証明書を作成します。次に例を示します。

```
# openssl req -sha256 -new -key server.key -out server.csr
```

ステップ 3 CSR を認証局に送信して、証明書（server.crt など）を取得します。

（注） WAE は、PEM フォーマットの server.crt のみをサポートします。サーバー証明書を DER から PEM フォーマットに変換するには、次のコマンドを使用できます。

```
sudo openssl x509 -inform der -in <input certificate filename> -out <output certificate filename>
```

ステップ 4 server.key および server.crt ファイルの場所を示すように <key-file/> および <cert-file/> 要素を変更し、*<WAE_run_directory>*/wae.conf を変更します。

ステップ 5 Cisco WAE サーバを再起動します。

```
# sudo supervisorctl stop wae:*
# sudo supervisorctl start wae:*
```

Cisco WAE Live の証明書のインストール

Cisco WAE Live には、証明書が信頼されていないことをブラウザに示す、デフォルトの証明書が含まれています。これは予期されている動作です。この警告は、適切な CA 発行の証明書を適用することで削除できます。

Cisco WAE Live の CA 証明書をインストールするには、次の手順を実行します。

始める前に

（注） この手順は、Cisco WAE Live 7.1.1 以降にのみ適用されます。

- このタスクを実行するには、Cisco WAE ユーザー権限を持つ管理者である必要があります。
- ツール keytool は、jdk/jre で展開されます。keytool path が PATH に含まれていることを確認してください。

（注） 前の例は、シェルが sh、ksh、または bash の場合に適用できます。
他のシェルには同等のコマンドを使用します。

- ログアウトして再度ログインするか、適切なプロファイルのファイル名を使用して次のコマンドを入力します。

  ```
  # source ~/.profile
  ```

ステップ 1 任意の認証局（CA）から証明書を取得するには、証明書署名要求（CSR）を作成する必要があります。CSR の作成手順は、次のとおりです。

a) デフォルトの証明書を削除します。次に例を示します。

```
# keytool -storepass changeit -delete -alias cisco -keystore
$CARIDEN_HOME/lib/web/apache-tomcat-8.5.53/conf/keystore
```

b) 自己署名証明書を作成します。次に例を示します。

```
# keytool -storepass changeit -genkey -alias tomcat -keyalg RSA -keystore
$CARIDEN_HOME/lib/web/apache-tomcat-8.5.53/conf/keystore
```

c) CSR を作成します。次に例を示します。

```
# keytool -storepass changeit -certreq -keyalg RSA -alias tomcat -file certreq.csr -keystore
$CARIDEN_HOME/lib/web/apache-tomcat-8.5.53/conf/keystore
```

d) CSR ファイルを認証局に提出して、証明書を取得します。

e) （オプション）Cisco WAE Live を再起動して、新しい証明書をすぐに使用します。

```
# embedded_web_server -action stop
# embedded_web_server -action start
```

ステップ 2 証明書をインストールします。

a) 証明書を取得したCAから、チェーン証明書（ルート証明書とも呼ばれます）をダウンロードします。

b) チェーン証明書をキーストアにインポートします。

```
# keytool -storepass changeit -import -alias root -keystore
$CARIDEN_HOME/lib/web/apache-tomcat-8.5.53/conf/keystore -trustcacerts -file
<filename_of_the_chain_certificate>
```

c) 新しい証明書をインポートします。

```
# keytool -storepass changeit -import -alias tomcat -keystore
$CARIDEN_HOME/lib/web/apache-tomcat-8.5.53/conf/keystore -file <your_certificate_filename>
```

d) Cisco WAE Live を再起動します。

```
# embedded_web_server -action stop
# embedded_web_server -action start
```

LDAP サーバーの証明書のインストール

Cisco WAE は、Lightweight Directory Access Protocol（LDAP）を使用して外部ユーザーの認証と許可をサポートしています。

LDAPS プロトコルを使用するには、SSL 証明書を取得してキーストアに追加します。

ステップ 1 次のコマンドを使用して、自己署名証明書を cert.pem ファイルに保存します。

```
# openssl s_client -connect <ldap-host>:<ldap-ssl-port> </dev/null 2>/dev/null | sed -n
'/^-----BEGIN/,/^-----END/ { p }' > cert.pem
```

ステップ 2 WAE_RUN ディレクトリから次のコマンドを実行して、デフォルトのキーストアパスを取得します。

```
# $WAE_ROOT/lib/exec/test-java-ssl-conn <ldap-host> <ldap-ssl-port> 2>1 | grep "trustStore is:"
```

上記のコマンドを実行すると、証明書が取得されるディレクトリを見つけるのに役立ちます。次のようなディレクトリになります。

```
trustStore is: /usr/lib/jvm/java-1.8.0-openjdk-1.8.0.102-4.b14.el7.x86_64/jre/lib/security/cacerts
```

ステップ 3 次のコマンドを使用して、証明書をデフォルトのキーストアにインポートします。

```
# sudo keytool -import -keystore <default-key-store-path> -storepass changeit -noprompt -file cert.pem
```

EPN-M サーバーの証明書のインストール

L1 収集に Cisco Evolved Programmable Network Manager（Cisco EPN Manager）エージェントを使用する場合は、証明書をインストールします。

ステップ 1 次のコマンドを使用して、自己署名証明書を cert.pem ファイルに保存します。

```
# openssl s_client -connect <epnm-host>:<epnm-port> </dev/null 2>/dev/null | sed -n
'/^-----BEGIN/,/^-----END/ { p }' > cert.pem
```

ステップ 2 次のコマンドを使用して、デフォルトのキーストアパスを取得します。通常、デフォルトのキーストアパスは /etc/pki/java/cacerts です。

```
# $WAE_ROOT/lib/exec/test-java-ssl-conn <epnm-host> <epnm-port> 2>1 | grep "trustStore is:"
```

ステップ 3 次のコマンドを使用して、証明書をデフォルトのキーストアにインポートします。

```
# sudo keytool -import -keystore <default-key-store-path> -storepass changeit -noprompt -file cert.pem
```

翻訳について
このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。