目次

概要

コンテンツ サービス スイッチ（CSS）の既存のキーと証明書がない場合、それらを CSS で生成できます。 CSS には、秘密キー、証明書署名要求（CSR）、および自己署名仮証明書を生成するプロセスを簡略化するための一連の証明書と秘密キーの管理ユーティリティが含まれています。 このドキュメントでは、認証局（CA）から新しい証明書を取得して、CSS にインストールする手順について説明します。

前提条件

要件

このドキュメントに関する固有の要件はありません。

使用するコンポーネント

このドキュメントは、特定のソフトウェアやハードウェアのバージョンに限定されるものではありません。

このドキュメントの情報は、特定のラボ環境にあるデバイスに基づいて作成されたものです。 このドキュメントで使用するすべてのデバイスは、クリアな（デフォルト）設定で作業を開始しています。 ネットワークが稼働中の場合は、コマンドが及ぼす潜在的な影響を十分に理解しておく必要があります。

表記法

ドキュメント表記の詳細は、『シスコ テクニカル ティップスの表記法』を参照してください。

設定

この項では、このドキュメントで説明する機能の設定に必要な情報を提供します。

注このドキュメントで使用されているコマンドの詳細を調べるには、Command Lookup Tool（登録ユーザ専用）を使用してください。

設定

このドキュメントでは、次の設定を使用します。

- Rivest、Shamir、Adelman (RSA) キー ペアの生成
- RSA キー ペア ファイルの関連付け
- CSR の生成
- Verisign 中間証明書の取得
- チェーン証明書ファイルのインポート
- 証明書ファイルの関連付け
- SSL プロキシ リストの設定
- セキュア ソケット レイヤ (SSL) サービスとコンテンツ ルールの設定

Rivest、Shamir、Adelman (RSA) キー ペアの生成

ssl genrsa コマンドを発行して、非対称暗号化用に RSA 秘密鍵/公開鍵キー ペアを生成します。 CSS は CSS でファイルとして生成された RSA キー ペアを保存します。 たとえば、RSA キー ペア myrsakey.pem を生成するには、次のように入力します。

```
CSS11500(config) # ssl genrsa myrsakey.pem 1024 ?passwd123?Please be patient this could take a few
minutes
```

RSA のキー ペア ファイルの関連付け

ssl associate rsakey コマンドを発行し、生成された RSA キー ペアに RSA キー ペア名を関連付けます。 たとえば、生成された RSA キー ペア ファイル myrsakey.pem に RSA キー名 myrsakey1 を関連付けるには、次のように入力します。

```
CSS11500(config) # ssl associate rsakey myrsakey1 myrsakey.pem
```

CSR の生成

ssl gencsr rsakey コマンドを発行し、関連付けをした RSA キー ペア ファイルの CSR ファイルを生成します。 この CSR は、署名用に CA に送信されます。 たとえば、RSA キー ペア myrsakey1 に基づいて CSR を生成するには、次のように入力します。

```
CSS11503(config)# ssl gencsr myrsakey1You are about to be asked to enter informationthat will be
incorporated into your certificaterequest. What you are about to enter is what iscalled a Distinguished
Name or a DN.For some fields there will be a default value,If you enter '.', the field will be left
blank.Country Name (2 letter code) [US] USState or Province (full name) [SomeState] CALocality Name
(city) [SomeCity] San JoseOrganization Name (company name) [Acme Inc]Cisco Systems, Inc.Organizational
Unit Name (section) [Web Administration] Web AdminCommon Name (your domain name) [www.acme.com]
www.cisco.comEmail address [webadmin@acme.com] webadmin@cisco.com
```

ssl gencsr コマンドにより、画面に CSR および出力が生成されます。 ほとんどの主要な CA には、画面に証明書リクエストをカット アンド ペーストすることを求める Web ベースのアプリケーションが実装されています。

```
CSS11503(config)# ssl gencsr myrsakey1You are about to be asked to enter informationthat will be
incorporated into your certificaterequest. What you are about to enter is what iscalled a Distinguished
Name or a DN.For some fields there will be a default value,If you enter '.', the field will be left
blank.Country Name (2 letter code) [US] USState or Province (full name) [SomeState] CALocality Name
(city) [SomeCity] San JoseOrganization Name (company name) [Acme Inc]Cisco Systems, Inc.Organizational
Unit Name (section) [Web Administration] Web AdminCommon Name (your domain name) [www.acme.com]
www.cisco.comEmail address [webadmin@acme.com] webadmin@cisco.com
```

CA によって、署名した CSR が返されます。この処理は、通常、CSR で提供される電子メールアドレスを使用して行われます。

Verisign 中間証明書の取得

CA からの証明書の取得

CA に CSR を送信した後、1 ~ 7 営業日の間に、署名済み証明書が届きます。 所要日数は、CA によって異なります。 CA が署名して提供した証明書は、CSS に追加できます。

ステップアップ/SGC またはチェーン証明書に適用する場合、証明書に署名する中間証明書を取得する必要があります。 次のリンクから VeriSign の中間証明書を取得できます。

- 中間 CA 証明書のインストール
中間証明書をファイルに保存します たとえば、intermediate.pem に保存します。

サーバ証明書と中間証明書の結合

CSS でチェーン証明書を使用するには、サーバ証明書と中間証明書を結合する必要があります。これにより、CSS は、最初の SSL ハンドシェイクにおいて、クライアントに完全な証明書チェーンを返すことができます。 CSS のチェーン証明書ファイルを作成する場合、証明書が正しい順序であることを確認します。 最初にサーバ証明書が配置される必要があり、サーバ証明書の署名に使用される中間証明書はその次に配置される必要があります。 サーバ証明書と中間証明書の間に、空白行を 1 行挿入します。 たとえば、サーバ証明書 servercert.pem と中間証明書 intermediate.pem を結合して、mychainedrsacert.pem というチェーン証明書を作成します。 次に、mychainedrsacert.pem の内容全体を示します。

```
CSS11503(config)# ssl gencsr myrsakey1You are about to be asked to enter informationthat will be
incorporated into your certificaterequest. What you are about to enter is what iscalled a Distinguished
Name or a DN.For some fields there will be a default value,If you enter '.', the field will be left
blank.Country Name (2 letter code) [US] USState or Province (full name) [SomeState] CALocality Name
(city) [SomeCity] San JoseOrganization Name (company name) [Acme Inc]Cisco Systems, Inc.Organizational
Unit Name (section) [Web Administration] Web AdminCommon Name (your domain name) [www.acme.com]
www.cisco.comEmail address [webadmin@acme.com] webadmin@cisco.com
```

チェーン証明書ファイルのインポート

CA によって CSR への署名が行われると、「証明書」と呼ばれるものが完成します。 証明書ファイルは CSS にインポートする必要があります。 **copy ssl コマンド**を発行して、CSS からの証明書および秘密鍵のインポート、または CSS への証明書および秘密鍵のエクスポートを実行します。 CSS は、インポートされたすべてのファイルを、CSS 上の安全な場所に保存します。 このコマンドは、SuperUser モードに限り使用できます。 たとえば、リモート サーバから CSS に mychainedrsacert.pem 証明書をインポートするには、次のように入力します。

```
CSS11500# copy ssl sftp ssl_record import mychainedrsacert.pem PEM ?passwd123?Connecting Completed
successfully
```

証明書ファイルの関連付け

ssl associate cert コマンドを発行して、証明書名をインポートされた証明書に関連付けます。 たとえば、証明書名 mychainedrsacert1 をインポートされた証明書ファイル mychainedrsacert.pem に関連付けるには、次のように入力します。

```
CSS11500(config)# ssl associate cert mychainedrsacert1 mychainedrsacert.pem
```

SSL プロキシ リストの設定

ssl-proxy-list コマンドを発行し、SSL プロキシ リストを変更します。 SSL プロキシ リストは、SSL サービスに関連付けられている、関連する仮想 SSL サーバまたはバックエンド SSL サーバのグループです。 SSL プロキシ リストには、各仮想 SSL サーバに関するすべての設定情報が含まれます。 この情報には、SSL サーバの作成、証明書および対応する SSL キー ペア、仮想 IP (VIP) アドレスおよびポート、サポートされている SSL 暗号化、および他の SSL オプションが含まれます。 たとえば、ssl-proxy-list ssl_list1 を作成するには、次のように入力します。

```
CSS11500(config)# ssl-proxy-list ssl_list1Create ssl-list <ssl_list1>, [y/n]: y
```

SSL プロキシ リストを作成すると、CLI は ssl-proxy-list コンフィギュレーション モードになります。 次に示すように、SSL サーバを設定します。

```
CSS11500(ssl-proxy-list[ssl_list1])# ssl-server 20CSS11500(ssl-proxy-list[ssl_list1])# ssl-server 20 vip address 192.168.3.6CSS11500(ssl-proxy-list[ssl_list1])# ssl-server 20 rsacert mychainedrsacert1CSS11500(ssl-proxy-list[ssl_list1])# ssl-server 20 rsakey myrsakey1CSS11500(ssl-proxy-list[ssl_list1])# ssl-server 20 cipher rsa-export-with-rc4-40-md5 192.168.11.2 80 5CSS11500(ssl-proxy-list[ssl_list1])# active
```

セキュア ソケット レイヤ (SSL) サービスとコンテンツ ルールの設定

SSL プロキシ リストが有効になったら、サービスとコンテンツ ルールを設定して、CSS が SSL モジュールに SSL トラフィックを送信できるようにする必要があります。 次の表に、仮想 SSL サーバ用に SSL サービスを作成するための手順 (概要) を示します。SSL プロキシ リストをサービスに追加する方法や、SSL コンテンツ ルールを作成する方法も記載しています。

SSL サービスの作成

```
CSS11500(config)# service ssl_serv1Create service <ssl_serv1>,    [y/n]: yCSS11500(config-service[ssl_serv1])# type ssl-accelCSS11500(config-service[ssl_serv1])# slot 2CSS11500(config-service[ssl_serv1])# keepalive type noneCSS11500(config-service[ssl_serv1])# add ssl-proxy-list ssl_list1CSS11500(config-service[ssl_serv1])# active
```

SSL コンテンツ ルールの作成

```
CSS11500(config)# owner ssl_ownerCreate owner <ssl_owner>, [y/n]: yCSS11500(config-owner[ssl_owner])# content ssl_rule1Create content <ssl_rule1>, [y/n]: yCSS11500(config-owner-content[ssl-rule1]# vip address 192.168.3.6CSS11500(config-owner-content[ssl-rule1]# port 443 CSS11500(config-owner-content[ssl_rule1])# add service ssl_serv1 CSS11500(config-owner-content[ssl_rule1])# active
```

クリア テキストのコンテンツ ルールの作成

```
CSS11500(config-owner[ssl_owner])# content decrypted_www Create content <decrypted_www>, [y/n]: yCSS11500(config-owner-content[decrypted_www]# vip address 192.168.11.2CSS11500(config-owner-content[decrypted_www]# port 80CSS11500(config-owner-content[decrypted_www])# add service linux_httpCSS11500(config-owner-content[decrypted_www])# add service win2k_httpCSS11500(config-owner-content[decrypted_www])# active
```

この時点で、クライアント HTTPS トラフィックは 192.168.3.6:443 にある CSS に送信できます。 CSS は、HTTPS トラフィックを復号化し、HTTP に変換します。 CSS ではサービスを選択し、HTTP Web サーバに HTTP トラフィックを送信します。 次に、上記の例を使用して動作している CSS の設定を示します。

```
CSS11501# show runconfigure!**************************** GLOBAL ****************************ssl associate rsakey myrsakey1 myrsakey.pemssl associate cert mychainedrsacert1 mychainedrsacert.pemip route 0.0.0.0 0.0.0.0 192.168.3.1 1ftp-record conf 192.168.11.101 admin des-password 4f2bxansrcehjgka /tftpboot!************************ INTERFACE ************************interface 1/1bridge vlan 10description "Client Side"interface 1/2bridge vlan 20description "Server Side"!************************ CIRCUIT ************************circuit VLAN10description "Client Segment"ip address 192.168.3.254 255.255.255.0circuit VLAN20description "Server Segment"ip address
```

```
192.168.11.1 255.255.255.0!********************** SSL PROXY LIST **********************ssl-proxy-list
ssl_list1ssl-server 20ssl-server 20 vip address 192.168.3.6ssl-server 20 rsakey myrsakey1ssl-server 20
rsacert mycertcert1ssl-server 20 cipher rsa-with-rc4-128-md5 192.168.11.2
80active!************************* SERVICE *************************service linux-httpip address
192.168.11.101port 80activeservice win2k-httpip address 192.168.11.102port 80activeservice ssl_serv1type
ssl-accelslot 2keepalive type noneadd ssl-proxy-list ssl_list1active!************************** OWNER
*************************owner ssl_ownercontent ssl_rule1vip address 192.168.3.6protocol tcpport 443add
service ssl_serv1activecontent decrypted_wwwvip address 192.168.11.2add service linux-httpadd service
win2k-httpprotocol tcpport 80active
```

確認

ここでは、設定が正常に動作していることを確認します。

show ssl file および **show ssl associate コマンド**を使用して、設定を確認します。

すべてのファイルのサイズが 0 よりも大きいことを確認します。

clear ssl file コマンドを使用して、証明書または鍵を削除できます。

トラブルシューティング

ここでは、設定に関するトラブルシューティングについて説明します。

SSL ネゴシエーションが失敗した場合、**show ssl statistics コマンド**を使用して、失敗した SSL ネゴシエーションに関する有益な情報を表示します。

たとえば、次のフィールドをチェックします。

```
CSS11501# show runconfigure!*************************** GLOBAL ***************************ssl associate
rsakey myrsakey1 myrsakey.pemssl associate cert mychainedrsacert1 mychainedrsacert.pemip route 0.0.0.0
0.0.0.0 192.168.3.1 1ftp-record conf 192.168.11.101 admin des-password 4f2bxansrcehjgka
/tftpboot!************************* INTERFACE *************************interface 1/1bridge vlan
10description "Client Side"interface 1/2bridge vlan 20description "Server
Side"!************************* CIRCUIT *************************circuit VLAN10description "Client
Segment"ip address 192.168.3.254 255.255.255.0circuit VLAN20description "Server Segment"ip address
192.168.11.1 255.255.255.0!********************** SSL PROXY LIST **********************ssl-proxy-list
ssl_list1ssl-server 20ssl-server 20 vip address 192.168.3.6ssl-server 20 rsakey myrsakey1ssl-server 20
rsacert mycertcert1ssl-server 20 cipher rsa-with-rc4-128-md5 192.168.11.2
80active!************************* SERVICE *************************service linux-httpip address
192.168.11.101port 80activeservice win2k-httpip address 192.168.11.102port 80activeservice ssl_serv1type
ssl-accelslot 2keepalive type noneadd ssl-proxy-list ssl_list1active!************************** OWNER
*************************owner ssl_ownercontent ssl_rule1vip address 192.168.3.6protocol tcpport 443add
service ssl_serv1activecontent decrypted_wwwvip address 192.168.11.2add service linux-httpadd service
win2k-httpprotocol tcpport 80active
```

関連情報

- CSS 11500
- CSS 11000 シリーズ コンテント サービス スイッチのハードウェア サポート (英語)
- Cisco WebNS CSS11500 ソフトウェアのダウンロード (登録ユーザ専用)
- Cisco WebNS CSS11000 ソフトウェアのダウンロード (登録ユーザ専用)
- テクニカルサポートとドキュメント - Cisco Systems