SSH 認証の X.509v3 証明書

SSH 認証の X.509v3 証明書機能は、公開キー アルゴリズム（PKI）を使用してサーバおよびユーザの認証を行い、認証局（CA）が署名し発行したデジタル証明書を介してキー ペアの所有者のアイデンティティをセキュア シェル（SSH）プロトコルによって検証することを可能します。

このモジュールでは、デジタル証明書用のサーバおよびユーザ証明書プロファイルを設定する方法について説明します。

機能情報の確認

ご使用のソフトウェア リリースでは、このモジュールで説明されるすべての機能がサポートされているとは限りません。最新の機能情報および警告については、使用するプラットフォームおよびソフトウェア リリースの Bug Search Tool およびリリース ノートを参照してください。このモジュールに記載されている機能の詳細を検索し、各機能がサポートされているリリースのリストを確認する場合は、このモジュールの最後にある機能情報の表を参照してください。

プラットフォームのサポートおよびシスコ ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator には、http://www.cisco.com/go/cfn からアクセスします。Cisco.com のアカウントは必要ありません。

SSH 認証の X.509v3 証明書の前提条件

SSH 認証の X.509v3 証明書機能では、**ipsshserverauthenticateuser** コマンドが **ipsshserveralgorithmauthentication** コマンドに置き換えられます。**ipsshserverauthenticateuser** コマンドを設定から削除するには、**defaultipsshserverauthenticateuser** コマンドを設定します。こうすると、IOS セキュア シェル（SSH）サーバが **ipsshserveralgorithmauthentication** コマンドを使用するようになります。

ipsshserverauthenticateuser コマンドを設定すると、次のメッセージが表示されます。

警告 SSH コマンドを受け入れました。ただし、この CLI はまもなく廃止されます。新しい CLI **ipsshserveralgorithmauthentication** に移行してください。CLI を無効にするには「**defaultipsshserverauthenticateuser**」を設定してください。

SSH 認証の X.509v3 証明書の制約事項

- SSH 認証の X.509v3 証明書機能の実装は、Cisco IOS セキュア シェル（SSH）サーバ側にのみ適用できます。
- Cisco IOS SSH サーバは、サーバおよびユーザ認証について、x509v3-ssh-rsa アルゴリズム ベースの証明書のみをサポートします。

SSH 認証用の X.509v3 証明書に関する情報

SSH 認証の X.509v3 証明書の概要

セキュア シェル（SSH）プロトコルは、ネットワーク デバイスへの安全なリモート アクセス接続を提供します。クライアントとサーバの間の通信は暗号化されます。

公開キー暗号化を使用して認証を行う SSH プロトコルが 2 つあります。トランスポート層プロトコルは、デジタル署名アルゴリズム（公開キーアルゴリズムと呼ばれます）を使用して、サーバをクライアントに対して認証します。一方、ユーザ認証プロトコルは、デジタル署名を使用して、クライアントをサーバに対して認証します（公開キー認証）。

認証の有効性は、公開署名キーとその署名者のアイデンティティとの関連の強さに依存します。X.509 バージョン 3（X.509v3）などのデジタル証明書は、アイデンティティ管理のために使用されます。X.509v3 は、信頼できるルート認証局とその中間認証局による署名の連鎖を使用して、公開署名キーを特定のデジタル アイデンティティにバインドします。この実装により、公開キー アルゴリズムを使用したサーバとユーザの認証が可能になるとともに、認証局

（CA）が署名し発行したデジタル証明書を介してキー ペアの所有者のアイデンティティを SSH で検証することが可能になります。

X.509v3 を使用したサーバおよびユーザ認証

サーバ認証の場合、セキュア シェル（SSH）サーバが確認のためにそれ自体の証明書を SSH クライアントに送信します。このサーバ証明書は、サーバ証明書プロファイル（ssh-server-cert-profile-server コンフィギュレーション モード）で設定されたトラストポイントに関連付けられます。

ユーザ認証の場合、SSH クライアントが確認のためにユーザの証明書を IOS SSH サーバに送信します。SSH サーバは、サーバ証明書プロファイル（ssh-server-cert-profile-user コンフィギュレーション モード）で設定された公開キー インフラストラクチャ（PKI）トラストポイントを使用して、受信したユーザ証明書を確認します。

デフォルトでは、証明書ベースの認証が、IOS SSH サーバ端末でサーバおよびユーザに対して有効になっています。

OCSP 応答ステープリング

オンライン証明書ステータス プロトコル（OCSP）では、識別された証明書の（失効）状態をアプリケーションが判断することが可能です。このプロトコルは、証明書のステータスをチェックするアプリケーションとそのステータスを提供するサーバとの間でやり取りする必要があるデータを指定します。OCSP クライアントは OCSP レスポンダにステータス要求を発行し、応答を受信するまで証明書の受け入れを保留します。OCSP 応答には、少なくとも、要求の処理ステータスを示す responseStatus フィールドが含まれます。

公開キー アルゴリズムの場合、キーの形式は、1 つ以上の X.509v3 証明書のシーケンスと、その後に続く 0 個以上の OCSP 応答のシーケンスから成ります。

SSH 認証機能向けの X.509v3 証明書は、OCSP 応答ステープリングを使用します。OCSP 応答ステープリングを使用することにより、デバイスは、OCSP サーバにアクセスしてから結果を証明書とともにステープリングして、ピアから OCSP レスポンダにアクセスさせるのではなくピアに情報を送ることで、自身の証明書の失効情報を取得します。

SSH 認証用の X.509v3 証明書の設定方法

サーバ認証用のデジタル証明書の設定

手順の概要

1. **enable**
2. **configureterminal**
3. **ip ssh server algorithm hostkey** {**x509v3-ssh-rsa** [**ssh-rsa**] | **ssh-rsa** [**x509v3-ssh-rsa**]}
4. **ip ssh server certificate profile**

5. **server**
6. **trustpoint sign** *PKI-trustpoint-name*
7. **ocsp-response include**
8. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Switch> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configureterminal** 例： `Switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip ssh server algorithm hostkey** {**x509v3-ssh-rsa** [**ssh-rsa**] | **ssh-rsa** [**x509v3-ssh-rsa**]} 例： `Switch(config)# ip ssh server algorithm hostkey x509v3-ssh-rsa`	ホスト キー アルゴリズムの順序を定義します。セキュア シェル（SSH）クライアントとネゴシエートされるのは、設定済みのアルゴリズムのみです。 （注） IOS SSH サーバには、1 つ以上の設定済みホスト キー アルゴリズムが必要です。 • **x509v3-ssh-rsa**：証明書ベースの認証 • **ssh-rsa**：公開キーベースの認証
ステップ 4	**ip ssh server certificate profile** 例： `Switch(config)# ip ssh server certificate profile`	サーバ証明書プロファイルおよびユーザ証明書プロファイルを設定し、SSH 証明書プロファイル コンフィギュレーション モードを開始します。
ステップ 5	**server** 例： `Switch(ssh-server-cert-profile)# server`	サーバ証明書プロファイルを設定し、SSH サーバ証明書プロファイルのユーザ コンフィギュレーション モードを開始します。 • サーバ プロファイルは、サーバ認証時にサーバの証明書を SSH クライアントに送信するために使用されます。
ステップ 6	**trustpoint sign** *PKI-trustpoint-name* 例： `Switch(ssh-server-cert-profile-server)# trustpoint sign trust1`	公開キー インフラストラクチャ（PKI）トラストポイントをサーバ証明書プロファイルにアタッチします。 • SSH サーバは、この PKI トラストポイントに関連付けられた証明書をサーバ認証に使用します。

	コマンドまたはアクション	目的
ステップ 7	**ocsp-response include** 例： `Switch(ssh-server-cert-profile-server)# ocsp-response include`	（任意）Online Certificate Status Protocol（OCSP）の応答または OCSP ステープリングをサーバ証明書と一緒に送信します。 （注） デフォルトでは、OCSP 応答はサーバ証明書と一緒には送信されません。
ステップ 8	**end** 例： `Switch(ssh-server-cert-profile-server)# end`	SSH サーバ証明書プロファイルのサーバ コンフィギュレーションモードを終了し、特権 EXEC モードに戻ります。

ユーザ認証用のデジタル証明書の設定

手順の概要

1. **enable**
2. **configureterminal**
3. **ip ssh server algorithm authentication** {**publickey** | **keyboard** | **password**}
4. **ip ssh server algorithm publickey** {**x509v3-ssh-rsa** [**ssh-rsa**] | **ssh-rsa** [**x509v3-ssh-rsa**]}
5. **ip ssh server certificate profile**
6. **user**
7. **trustpoint verify** *PKI-trustpoint-name*
8. **ocsp-response required**
9. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Switch> enable`	特権 EXEC モードをイネーブルにします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configureterminal** 例： `Switch# configure terminal`	グローバル コンフィギュレーション モードを開始します。
ステップ 3	**ip ssh server algorithm authentication** {**publickey**	**keyboard**	**password**} 例：	ユーザ認証アルゴリズムの順序を定義します。セキュア シェル（SSH）クライアントとネゴシエートされるのは、設定済みのアルゴリズムのみです。

	コマンドまたはアクション	目的
	`Switch(config)# ip ssh server algorithm authentication publickey`	（注） • IOS SSH サーバには、1 つ以上の設定済みユーザ認証アルゴリズムが必要です。 • ユーザ認証に証明書方式を使用するには、**publickey** キーワードを設定する必要があります。
ステップ 4	**ip ssh server algorithm publickey** {**x509v3-ssh-rsa** [**ssh-rsa**] | **ssh-rsa** [**x509v3-ssh-rsa**]} 例： `Switch(config)# ip ssh server algorithm publickey x509v3-ssh-rsa`	公開キーアルゴリズムの順序を定義します。SSH クライアントによってユーザ認証に許可されるのは、設定済みのアルゴリズムのみです。 （注） IOS SSH クライアントには、1 つ以上の設定済み公開キーアルゴリズムが必要です。 • **x509v3-ssh-rsa**：証明書ベースの認証 • **ssh-rsa**：公開キーベースの認証
ステップ 5	**ip ssh server certificate profile** 例： `Switch(config)# ip ssh server certificate profile`	サーバ証明書プロファイルおよびユーザ証明書プロファイルを設定し、SSH 証明書プロファイル コンフィギュレーション モードを開始します。
ステップ 6	**user** 例： `Switch(ssh-server-cert-profile)# user`	ユーザ証明書プロファイルを設定し、SSH サーバ証明書プロファイルのユーザ コンフィギュレーション モードを開始します。
ステップ 7	**trustpoint verify** *PKI-trustpoint-name* 例： `Switch(ssh-server-cert-profile-user)# trustpoint verify trust2`	受信したユーザ証明書の確認に使用される公開キーインフラストラクチャ（PKI）トラストポイントを設定します。 （注） 同じコマンドを複数回実行することで、複数のトラストポイントを設定します。最大 10 のトラストポイントを設定できます。
ステップ 8	**ocsp-response required** 例： `Switch(ssh-server-cert-profile-user)# ocsp-response required`	（任意）受信したユーザ証明書による Online Certificate Status Protocol（OCSP）の応答の有無を要求します。 （注） デフォルトでは、ユーザ証明書は OCSP 応答なしで受け入れられます。
ステップ 9	**end** 例： `Switch(ssh-server-cert-profile-user)# end`	SSH サーバ証明書プロファイルのユーザ コンフィギュレーション モードを終了し、特権 EXEC モードに戻ります。

デジタル証明書を使用したサーバおよびユーザ認証の確認

手順の概要

1. **enable**
2. **show ip ssh**

手順の詳細

ステップ 1 **enable**

特権 EXEC モードをイネーブルにします。

- パスワードを入力します（要求された場合）。

例：

```
Device> enable
```

ステップ 2 **show ip ssh**

現在設定されている認証方式を表示します。証明書ベース認証の使用を確認するには、x509v3-ssh-rsa アルゴリズムが設定済みのホスト キー アルゴリズムであることを確認します。

例：

```
Device# show ip ssh

SSH Enabled - version 1.99
Authentication methods:publickey,keyboard-interactive,password
Authentication Publickey Algorithms:x509v3-ssh-rsa,ssh-rsa
Hostkey Algorithms:x509v3-ssh-rsa,ssh-rsa
Authentication timeout: 120 secs; Authentication retries: 3
Minimum expected Diffie Hellman key size : 1024 bits
```

SSH 認証用の X.509v3 証明書の設定例

例：サーバ認証用のデジタル証明書の設定

```
Switch> enable
```

```
Switch# configure terminal
Switch(config)# ip ssh server algorithm hostkey x509v3-ssh-rsa
Switch(config)# ip ssh server certificate profile
Switch(ssh-server-cert-profile)# server
Switch(ssh-server-cert-profile-server)# trustpoint sign trust1
Switch(ssh-server-cert-profile-server)# exit
```

例：ユーザ認証用のデジタル証明書の設定

```
Switch> enable
Switch# configure terminal
Switch(config)# ip ssh server algorithm authentication publickey
Switch(config)# ip ssh server algorithm publickey x509v3-ssh-rsa
Switch(config)# ip ssh server certificate profile
Switch(ssh-server-cert-profile)# user
Switch(ssh-server-cert-profile-user)# trustpoint verify trust2
Switch(ssh-server-cert-profile-user)# end
```

SSH 認証の X.509v3 証明書に関するその他の参考資料

関連資料

関連項目	マニュアル タイトル
PKI 設定	PKI 展開での Cisco IOS 証明書サーバの設定および管理

シスコのテクニカル サポート

説明	Link
シスコのサポート Web サイトでは、シスコの製品やテクノロジーに関するトラブルシューティングにお役立ていただけるように、マニュアルやツールをはじめとする豊富なオンライン リソースを提供しています。 お使いの製品のセキュリティ情報や技術情報を入手するために、Cisco Notification Service（Field Notice からアクセス）、Cisco Technical Services Newsletter、Really Simple Syndication（RSS）フィードなどの各種サービスに加入できます。 シスコのサポート Web サイトのツールにアクセスする際は、Cisco.com のユーザ ID およびパスワードが必要です。	http://www.cisco.com/support

SSH 認証の X.509v3 証明書の機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

プラットフォームのサポートおよびシスコ ソフトウェア イメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、www.cisco.com/go/cfn に移動します。Cisco.com のアカウントは必要ありません。

表 *1 : SSH* 認証の *X.509v3* 証明書の機能情報

機能名	リリース	機能情報
SSH 認証の X.509v3 証明書	Cisco IOS 15.2(4)E1	SSH 認証の X.509v3 証明書機能は、サーバ内で X.509v3 デジタル証明書を使用し、SSH サーバ側でユーザ認証を使用します。 次のコマンドが導入または変更されました。**ip ssh server algorithm hostkey**、**ip ssh server algorithm authentication**、および **ip ssh server certificate profile**。 この機能は、次のプラットフォームに実装されていました。 • Catalyst 2960C、2960CX、2960P、2960X、および 2960XR シリーズ スイッチ • Catalyst 3560CX および 3560X シリーズ スイッチ • Catalyst 3750X シリーズ スイッチ • Catalyst 4500E Sup7-E、Sup7L-E、Sup8-E および 4500X シリーズ スイッチ • Catalyst 4900M、4900F-E シリーズ スイッチ