



SSH 認証の X.509v3 証明書

- [SSH 認証用の X.509v3 証明書の機能履歴, on page 1](#)
- [SSH 認証の X.509v3 証明書, on page 2](#)
- [SSH 認証用 X.509v3 証明書の機能, on page 2](#)
- [SSH 認証の X.509v3 証明書の前提条件, on page 3](#)
- [SSH 認証の X.509v3 証明書の制約事項, on page 3](#)
- [SSH 認証用の X.509v3 証明書の設定方法, on page 3](#)
- [ユーザー認証用のデジタル証明書を確認するための SSH サーバーの設定, on page 5](#)
- [トラストポイント認証の設定とデバイス証明書の作成, on page 7](#)
- [デジタル証明書を使用したサーバーユーザー認証の設定確認, on page 9](#)
- [SSH 認証用の X.509v3 証明書の設定例, on page 10](#)

SSH 認証用の X.509v3 証明書の機能履歴

次の表に、このモジュールで説明する機能のリリース情報とプラットフォームサポート情報を示します。

これらの機能は、特に明記されていない限り、導入されたリリース以降のすべてのリリースで使用できます。

リリース	機能名と説明	サポートされるプラットフォーム
Cisco IOS XE 26.2.1ea	SSH 認証用 X.509v3 証明書: SSH 認証用 X.509v3 証明書の機能サポートが導入されました。	Cisco C9550 シリーズ スマートスイッチ

リリース	機能名と説明	サポートされるプラットフォーム
Cisco IOS XE 17.18.1	SSH 認証用 X.509v3 証明書: SSH 認証用 X.509v3 証明書により、SSH の証明書ベースのサーバーとユーザー認証が可能になり、PKI トラストポイントと X.509v3 デジタル証明書が ID 管理の強化に利用されます。	Cisco C9350 シリーズ スマートスイッチ Cisco C9610 シリーズ スマートスイッチ

SSH 認証の X.509v3 証明書

セキュアシェル（SSH）認証用の X.509v3 証明書機能は、サーバーと SSH サーバー側のユーザー認証の両方で X.509v3 デジタル証明書を使用できるようにします。このモジュールでは、これらの認証タイプの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。

SSH 認証用 X.509v3 証明書の機能

デジタル証明書

X.509v3 形式（RFC5280）のデジタル証明書は、ID 管理を提供します。信頼できるルートおよび中間認証局（CA）からの署名のチェーンを介して、公開署名キーと署名者のアイデンティティ間のリンクが確立されます。公開キーインフラストラクチャ（PKI）のトラストポイントは、デジタル証明書の管理に役立ちます。トラストポイントには、CA、アイデンティティパラメータ、およびデジタル証明書に関する情報が含まれており、追跡とさまざまな証明書との関連付けができます。

サーバーとユーザー認証

デフォルトでは、証明書ベースの認証が SSH サーバーでサーバーおよびユーザーに対して有効になります。

- **サーバー認証:** Cisco IOS XE SSH サーバー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。このサーバー証明書は、`conf-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。
- **ユーザー認証:** SSH クライアントが確認のためにユーザーの証明書を SSH サーバーに送信します。SSH サーバーは、`conf-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 PKI トラストポイントを使用して、受信したユーザー証明書を確認します。

SSH 認証の X.509v3 証明書の前提条件

廃止された **ip ssh server authenticate user** コマンドは、**ip ssh server algorithm authentication** コマンドに置き換わります。古いコマンドを使用すると、次の警告メッセージが表示されます。

```
Warning: SSH command accepted but this CLI will be deprecated soon.  
Please move to new CLI "ip ssh server algorithm authentication".  
Please configure "default ip ssh server authenticate user" to make the CLI  
ineffective.
```

古いコマンドを無効にして、SSH サーバーが新しいコマンド **default ip ssh server authenticate user** を使用するようにします。

SSH 認証の X.509v3 証明書の制約事項

- この機能は、Cisco IOS XE セキュアシェル（SSH）サーバー側にのみ適用されます。
- SSH サーバーは、サーバーおよびユーザー認証について、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 証明書の設定方法

サーバー認証にデジタル証明書を使用するように SSH サーバーを設定するには、次の手順を実行します。

Procedure

Step 1 enable

Example:

```
Device> enable
```

特権 EXEC モードを有効にします。プロンプトが表示されたらパスワードを入力します。

Step 2 configure terminal

Example:

```
Device# 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。

Step 3 ip ssh server algorithm hostkey {x509v3-ssh-rsa [ssh-rsa] | ssh-rsa [x509v3-ssh-rsa]}

Example:

```
Device(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。

Note

IOS SSH サーバーには、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。

- ssh-rsa: 公開キーベース認証
- x509v3-ssh-rsa: 証明書ベース認証

Step 4 ip ssh server certificate profile

Example:

```
Device(config)# ip ssh server certificate profile
```

サーバー証明書プロファイルおよびユーザー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。

Step 5 server

Example:

```
Device(conf-ssh-server-cert-profile)# server
```

サーバー証明書プロファイルを設定し、SSH サーバー証明書プロファイルのユーザー コンフィギュレーション モードを開始します。

Step 6 trustpoint sign PKI-trustpoint- name

Example:

```
Device(conf-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。

Step 7 ocsp-response include

Example:

```
Device(conf-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。

Note

デフォルトではこのコマンドの no 形式が設定されており、OCSP 応答はサーバー証明書と一緒に送信されません。

Step 8 end

Example:

```
Device(conf-ssh-server-cert-profile-server)# end
```

SSH サーバー証明書プロファイルのサーバー 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。

ユーザー認証用のデジタル証明書を確認するための SSH サーバーの設定

ユーザー認証にデジタル証明書を使用するように SSH サーバーを設定するには、次の手順を実行します。

Procedure

Step 1 enable

Example:

```
Device> enable
```

特権 EXEC モードを有効にします。プロンプトが表示されたらパスワードを入力します。

Step 2 configure terminal

Example:

```
Device# 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。

Step 3 ip ssh server algorithm authentication {publickey | keyboard | password}

Example:

```
Device(config)# ip ssh server algorithm authentication publickey
```

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。

Note

- SSH サーバーには、1 つ以上の設定済みユーザー認証アルゴリズムが必要です。
- ユーザー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。
- **ip ssh server algorithm authentication** コマンドは **ip ssh server authenticate user** コマンドの代わりに使用します。

Step 4 ip ssh server algorithm publickey {x509v3-ssh-rsa [ssh-rsa] | ssh-rsa [x509v3-ssh-rsa]}

Example:

```
Device(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。

Note

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。

- ssh-rsa: 公開キーベース認証

- x509v3-ssh-rsa: 証明書ベース認証

Step 5 ip ssh server certificate profile

Example:

```
Device(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。

Step 6 user

Example:

```
Device(conf-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。

Step 7 trustpoint verify PKI-trustpoint-name

Example:

```
Device(conf-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。

Note

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。

Step 8 ocsp-response required

Example:

```
Device(conf-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。

Note

デフォルトではこのコマンドの no 形式が設定されており、ユーザー証明書は OCSP 応答なしで受け入れられます。

Step 9 end

Example:

```
Device(conf-ssh-server-cert-profile-user)# end
```

SSH サーバ証明書プロファイルのユーザー 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。

トラストポイント認証の設定とデバイス証明書の作成

トラストポイント認証を設定してデバイス証明書を作成するには、次の手順を実行します。

**Note**

新しく設定した PKI 証明書には、新しい RSA キーペア名を使用することをお勧めします。既存の RSA キーペア名（古い証明書に関連付けられている）を新しい PKI 証明書に再利用する場合は、次のいずれかを実行します。

- 既存の RSA キーペア名を使用して新しい RSA キーペアを再生成するのではなく、既存の RSA キーペア名を再利用します。既存の RSA キーペア名を使用して新しい RSA キーペアを再生成すると、既存の RSA キーペアに関連付けられているすべての証明書が無効になります。
- まず古い PKI 証明書設定を手動で削除してから、新しい PKI 証明書に既存の RSA キーペア名を再利用します。

Procedure

Step 1**enable****Example:**

```
Device> enable
```

特権 EXEC モードを有効にします。プロンプトが表示されたらパスワードを入力します。

Step 2**configure terminal****Example:**

```
Device# 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。

Step 3**crypto pki trustpoint name****Example:**

```
Device(config)# crypto pki trustpoint trust1
```

トラストポイントおよび設定された名前を宣言して、CA トラストポイント コンフィギュレーション モードを開始します。

Step 4**enrollment url url****Example:**

```
Device(ca-trustpoint)# enrollment url http://10.1.1.10:80
```

デバイスが証明書要求を送信する CA の URL を指定します。

Step 5**revocation-check none**

Example:

```
Device(ca-trustpoint)# revocation-check none
```

証明書の確認が無視されることを指定します。

Step 6

rsakeypair *key-label* [*keysize* [*encryption-key-size*]]

Example:

```
Device(ca-trustpoint)# rsakeypair trust1 2048
```

（任意）証明書に関連付けるキーペアを指定します。key-label 引数付きのキーペアがまだ存在しない、あるいは **auto-enroll regenerate** コマンドが発行された場合、登録時に key-label 引数付きのキーペアが生成されます。

鍵を生成するための key-size 引数を指定し、encryption-keysize 引数を指定して、個別の暗号化、署名鍵、および証明書を要求します。key-size 引数の範囲は 512 ～ 4096 です。key-size と encryption-key-size は同じサイズでなければなりません。2048 未満の長さを指定することは推奨されません。

Note

このコマンドがイネーブルでない場合に、FQDN キーペアが使用されます。

Step 7

exit

Example:

```
Device(ca-trustpoint)# exit
```

CA トラストポイントコンフィギュレーションモードを終了し、グローバルコンフィギュレーションモードに戻ります。

Step 8

crypto pki authenticate *name*

Example:

```
Device(config)# crypto pki authenticate trust1
```

CA 証明書を取得して、認証します。証明書フィンガープリントをチェックするよう求められた場合、証明書フィンガープリントをチェックします。

Note

CA 証明書がコンフィギュレーションにすでにロードされている場合、このコマンドはオプションです。

Step 9

crypto pki enroll *name*

Example:

```
Device(config)# crypto pki enroll trust1
```

証明書要求が証明書サーバーに送信され、サーバーが ID またはデバイス証明書を発行します。証明書要求にルータの FQDN および IP アドレスを含めるかどうかなどの登録情報を求められます。

Step 10

show crypto pki certificates

Example:

```
Device# show crypto pki certificates verbose trust1
```

(任意) ロールオーバー証明書などの、証明書に関する情報を表示します。

What to do next

他の登録オプションを使用して証明書をインストールする方法の詳細については、「[PKI 内での RSA キーの展開](#)」を参照してください。

デジタル証明書を使用したサーバーユーザー認証の設定確認

デジタル証明書を使用したサーバーおよびユーザー認証の設定を確認するには、次の手順を実行します。

Procedure

Step 1 enable

Example:

```
Device> enable
```

特権 EXEC モードを有効にします。プロンプトが表示されたらパスワードを入力します。

Step 2 show ip ssh

Example:

```
Device# show ip ssh
SSH Enabled - version 2.0
Authentication
methods:publickey,keyboardinteractive,
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
```

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。

SSH 認証用の X.509v3 証明書の設定例

ここでは、デジタル証明書を使用したユーザーおよびサーバー認証の例を示します。

この例では、サーバー認証用のデジタル証明書を使用するための SSH サーバーの設定方法を示します。

```
Device> enable
Device# configure terminal
Device(config)# ip ssh server algorithm hostkey x509v3-ssh-rsa
Device(config)# ip ssh server certificate profile
Device(conf-ssh-server-cert-profile)# server
Device(conf-ssh-server-cert-profile-server)# trustpoint sign trust1
Device(conf-ssh-server-cert-profile-server)# end
```

この例では、ユーザー認証用のユーザーのデジタル証明書を確認するための SSH サーバーの設定方法を示します。

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

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。