



セキュア シェル

- [セキュアシェルの機能履歴, on page 1](#)
- [セキュアシェルとは, on page 2](#)
- [SSH の機能, on page 2](#)
- [SSH 設定時の注意事項, on page 3](#)
- [Strict-KEX, on page 4](#)
- [セキュア シェルを設定するための前提条件, on page 4](#)
- [セキュア シェルの設定に関する制約事項, on page 5](#)
- [セキュア シェルの設定, on page 6](#)
- [セキュアシェルの設定例, on page 9](#)

セキュアシェルの機能履歴

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: SSH 機能の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
Cisco IOS XE 17.18.1	SSH: SSH は、Telnet や Berkeley r-tools などの安全性の低いリモート接続ツールの暗号化された安全な代替手段を提供するアプリケーションおよびプロトコルです。	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

セキュアシェルとは

セキュアシェル（SSH）は、TelnetやBerkeley r-toolsなどの安全性の低いリモート接続ツールの暗号化された安全な代替手段を提供するアプリケーションおよびプロトコルです。

セッションを

- 標準暗号化メカニズムを使用して保護し、
- デバイス認証および暗号化を提供して、
- 保護されていないネットワークを介したセキュアな通信を可能にします。

SSH でサポートされている認証方式は次のとおりです。

- TACACS+
- RADIUS
- ローカル認証および許可

セキュアで認証および暗号化されたリモートアクセスを提供するために、SSHの実装には、プライマリアプリケーション、メカニズム、暗号化アルゴリズム、プロトコル拡張などの機能が含まれています。以降では、これらの機能について説明します。

SSH の機能

SSH サーバ



Note 特に明記しない限り、「SSH」という用語は「SSH バージョン 1」のみを意味します。

SSH サーバーの機能によって、SSH クライアントはシスコデバイスに対してセキュアで暗号化された接続を実行できます。インバウンド Telnet 接続と同様の機能が提供されますが、シスコソフトウェア認証で使用する強力な暗号化が可能です。Cisco ソフトウェアの SSH サーバーは、市販の一般的な SSH クライアントと相互運用できます。

SSH 統合クライアント

SSH 統合クライアントは、SSH プロトコル上で動作し、デバイス認証および暗号化を提供するアプリケーションです。シスコデバイスから SSH サーバーを実行している別のデバイスに対してセキュアで暗号化された接続を確立でき、アウトバウンド Telnet 接続と同様の機能で暗号化されます。シスコソフトウェアの SSH クライアントは、市販の一般的な SSH サーバーと使用します。

SSH クライアントでサポートされている暗号は次のとおりです。

- データ暗号規格（DES）
- 3DES

SSH クライアントはパスワード認証をサポートしています。



Note SSH クライアント機能を使用できるのは、SSH サーバーが有効になっている場合だけです。

SSH サーバー機能を使用すると、SSH クライアントはセキュアに接続できます。SSH 統合クライアント機能を使用すると、デバイスは別の SSH サーバーへのセキュアな接続を開始できます。両方が連携して、安全でない可能性のあるネットワーク上でセキュアな通信を確保します。

RSA 認証のサポート

セキュアシェル（SSH）クライアントで使用できる Rivest、Shamir、Adleman（RSA）認証は、シスコソフトウェアの SSH サーバではデフォルトでサポートされていません。

セキュアシェル（SSH）統合クライアント機能は、SSH プロトコル上で動作し、デバイスの認証および暗号化を実現するアプリケーションです。SSH クライアントによって、シスコデバイスは別のシスコデバイスなど SSH サーバを実行するデバイスに対して、セキュアで暗号化された接続を実行できます。この接続は、接続が暗号化される点を除いて Telnet のアウトバウンド接続と同様の機能を提供します。SSH クライアントは、認証および暗号化により、保護されていないネットワーク上でもセキュアな通信ができます。

SSH サーバおよび SSH 統合クライアントは、スイッチ上で実行されるアプリケーションです。SSH サーバは、このリリースでサポートされている SSH クライアントおよび、他社製の SSH クライアントと使用します。SSH クライアントは、市販の一般的な SSH サーバと連動します。SSH クライアントは、Data Encryption Standard（DES）、3DES、およびパスワード認証の暗号をサポートします。

ユーザ認証は、デバイスに対する Telnet セッションの認証と同様に実行されます。SSH は、次のユーザ認証方式もサポートします。

- TACACS+
- RADIUS
- ローカル認証および許可

SSH 設定時の注意事項

- SSHv2 サーバは、SSHv1 サーバで生成される RSA キーのペアを使用できます（逆の場合も同様です）。

- SSH サーバーがアクティブスイッチ上で動作しており、アクティブスイッチに障害が発生した場合、新しいアクティブスイッチは、以前のアクティブスイッチによって生成された RSA キーペアを使用します。
- 暗号キー生成 **rsa** コマンドを入力した後に CLI エラーメッセージが表示される場合は、ホスト名とドメインを再設定してから、もう一度コマンドを入力してください。
- RSA キーペアを生成するときに「No host name specified」というメッセージが表示される場合は、グローバル コンフィギュレーション モードで **hostname** コマンドを使用してホスト名を設定します。
- RSA キーペアを生成するときに「No domain specified」というメッセージが表示される場合は、グローバル コンフィギュレーション モードで **ip domain name** コマンドを使用して IP ドメイン名を設定します。
- ローカル認証および許可の方法を設定する場合に、コンソール上で AAA が無効になっていることを確認します。

Strict-KEX

Strict-KEX は、予期しないパケットやシーケンス外のパケットによって接続がすぐに切断されるようにし、キー交換後にシーケンス番号をリセットすることで、セキュリティを強化するプロトコル拡張です。継続的な保護を提供するために、strict-KEX はデフォルトで有効になっており、無効にすることはできません。

セキュア シェルを設定するための前提条件

セキュアで信頼性の高い操作を保証するために、セキュアシェル（SSH）を設定する前に、すべてのデバイスおよびソフトウェアの前提条件を満たしていることを確認します。

一般的な SSH 前提条件

デバイスで SSH またはセキュアコピープロトコル（SCP）を有効にする前に、次の要件を確認します。

- SSH を動作させるには、スイッチに Rivest、Shamir、および Adleman（RSA）の公開キーと秘密キーのペアが必要です。これは SSH が必要な Secure Copy Protocol（SCP）も同様に、セキュアな転送を実現させるには、これらのキーのペアが必要です。
- デバイスに必要なイメージをダウンロードします。セキュアシェル（SSH）サーバーには、IPsec（データ暗号規格（DES）または 3DES）の暗号化ソフトウェアイメージが必要です。SSH クライアントには、IPsec（DES または 3DES）の暗号化ソフトウェアイメージが必要です。
- グローバル コンフィギュレーション モードで **hostname** および **ip domain name** コマンドを使用して、デバイスのホスト名とホストドメインを設定します。

- デバイスの Rivest, Shamir、および Adleman (RSA) キーペアを生成します。このキーペアは、**crypto key generate rsa** コマンドを特権 EXEC モードで入力すると、自動的に SSH とリモート認証を有効にします。

認証および互換性の要件

適切な SSH 操作を確保するために、次の認証および互換性の要件を満たします。

- ローカルアクセスまたはリモートアクセス用にユーザー認証を設定します。認証、許可、アカウントिंग (AAA) の有無に関係なく、認証を設定できます。
- セキュアシェル (SSH) サーバーには、IPsec (データ暗号規格 (DES) または 3DES) の暗号化ソフトウェアイメージが必要です。SSH クライアントには、IPsec (DES または 3DES) の暗号化ソフトウェアイメージが必要です。
- Strict-KEX が動作するには、デバイスで SSH サポートが有効な Cisco IOS-XE を実行している必要があります。
- Strict-KEX が効果的に機能するには、クライアントデバイスとサーバーデバイスの両方が strict-kex アルゴリズムと互換性がある必要があります。

特記事項



Note 特に明記しない限り、「SSH」という用語は「SSH バージョン 1」のみを意味します。



Note RSA キーのペアを削除するには、**crypto key zeroize rsa** グローバル コンフィギュレーション コマンドを使用します。RSA キーペアを削除すると、SSH サーバーは自動的にディセーブルになります。

セキュア シェルの設定に関する制約事項

セキュアシェル (SSH) サーバーと SSH クライアントは、Data Encryption Standard (DES) (56 ビット) および 3DES (168 ビット) データ暗号化ソフトウェアイメージのみでサポートされます。DES ソフトウェア イメージの場合、使用できる暗号化アルゴリズムは DES だけです。3DES ソフトウェア イメージの場合、DES と 3DES の両方の暗号化アルゴリズムを使用できます。

- 実行シェルは、唯一サポートされるアプリケーションです。
- ログイン バナーはセキュア シェル バージョン 1 ではサポートされません。セキュア シェル バージョン 2 ではサポートされています。
- SFTP サーバーはサポートされていません。



Note 特に明記しない限り、「SSH」という用語は「SSH バージョン 1」のみを意味します。

セキュア シェルの設定

セキュアシェル（SSH）を設定して、デバイスへのセキュアなリモートアクセスを有効にし、管理セッションを不正アクセスから保護する方法を学習します。

SSH を実行するためのデバイスの設定

認証、ホスト名、ドメイン名を設定し、RSA キーを生成することにより、SSH を実行するようにデバイスを設定します。

この手順では、必要なすべての設定が行われていることを確認することにより、SSH を使用したデバイスへのセキュアなリモートアクセスが可能になります。

Before you begin

ローカルアクセスまたはリモートアクセス用にユーザ認証を設定します。この手順は必須です。

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 hostname *hostname*

Example:

```
Device(config)# hostname your_hostname
```

デバイスのホスト名および IP ドメイン名を設定します。

Note

この手順を実行するのは、デバイスを SSH サーバとして設定する場合だけです。

Step 4 ip domain name *domain_name*

Example:

```
Device(config)# ip domain name your_domain
```

デバイスのホストドメインを設定します。

Step 5 **exit**

Example:

```
Device(config)# exit
```

グローバル 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。

Step 6 **crypto key generate rsa**

Example:

```
Device# crypto key generate rsa
```

デバイス上でローカルおよびリモート認証用に SSH サーバをイネーブルにし、RSA キー ペアを生成します。デバイスの RSA キー ペアを生成すると、SSH が自動的にイネーブルになります。

最小モジュラス サイズは、1024 ビットにすることを推奨します。

RSA キーのペアを生成する場合に、モジュラスの長さの入力を求められます。モジュラスが長くなるほど安全ですが、生成と使用に時間がかかります。

Note

この手順を実行するのは、デバイスを SSH サーバとして設定する場合だけです。

Step 7 **show ip ssh**

Example:

```
Device# show ip ssh
```

(任意) SSH サーバーが有効であることを確認し、SSH 接続のバージョンおよび設定データを表示します。

SSH サーバーの設定



Note

特に明記しない限り、「SSH」という用語は「SSH バージョン 1」のみを意味します。

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 { time-out seconds | authentication-retries integer }**

Example:

```
Device(config)# ip ssh time-out 30
```

セキュアシェル（SSH）制御パラメータを設定します。

Note

Note このコマンドは、ユーザーに表示するパスワードプロンプトの回数を設定するためにも使用できます。この数値は、次の 2 つの値の低い方です。

- **ssh -o numberofpasswordprompt** コマンドを使用してクライアントから提案された値。
- **ip ssh authentication-retries integer** コマンドを使用してデバイスに設定された値に 1 を加えた値。

Step 4 **ip ssh rekey { time time | volume volume }**

Example:

```
Device(config)# ip ssh rekey time 108
```

（任意）SSH の時間ベースのキー再生成またはボリュームベースのキー再生成を設定します。

Step 5 **exit**

Example:

```
Device(config)# exit
```

グローバル 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。

Step 6 **show ip ssh**

Example:

```
Device# show ip ssh
```

（任意）SSH サーバーが有効であることを確認し、SSH 接続のバージョンおよび設定データを表示します。

SSH クライアントの呼び出し



Note 特に明記しない限り、「SSH」という用語は「SSH バージョン 1」のみを意味します。

セキュアシェル（SSH）クライアントを呼び出すには、次の手順を実行します。SSH クライアントはユーザー EXEC モードで実行されます。設定作業は特にありません。

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 `ssh -l username -vrf vrf-name ip-address`

Example:

```
Device# ssh -l user1 -vrf vrf1 192.0.2.1
```

SSH クライアントを呼び出し、指定した仮想ルーティングおよび転送（VRF）インスタンスの IP ホストまたはアドレスに接続します。

セキュアシェルの設定例

次の手順と設定例を使用して、デバイスでセキュアシェル（SSH）を設定する方法を学習します。

例：SSH サーバーの設定

次に、サーバーに設定されたセキュアシェル（SSH）制御パラメータの例を示します。この例では、30 秒のタイムアウト間隔が指定されています。このタイムアウト間隔は、SSH ネゴシエーションフェーズで使用されます。



Note 特に明記しない限り、「SSH」という用語は「SSH バージョン 1」のみを意味します。

Procedure

SSH サーバーの設定

```
Device> enable
Device# configure terminal
Device(config)# ip ssh timeout 30
Device(config)# end
```

例: SSH クライアントの呼び出し

次の例では、指定された Virtual Routing and Forwarding (VRF) インスタンスの IP アドレス 192.0.2.1 に接続するためにセキュアシェル (SSH) クライアントが呼び出されます。



Note 特に明記しない限り、「SSH」という用語は「SSH バージョン 1」のみを意味します。

Procedure

SSH クライアントを呼び出します。

```
Device> enable
Device# ssh -l user1 -vrf vrf1 192.0.2.1
```

SSH の確認



Note 特に明記しない限り、「SSH」という用語は「SSH バージョン 1」のみを意味します。

Procedure

Step 1 **show ip ssh** コマンドを使用して、SSH が有効になっていることを確認し、バージョンと設定データを表示します。

Example:

```
Device# show ip ssh
SSH Enabled - version 1.5
Authentication timeout: 120 secs; Authentication retries: 3
```

上記の出力は、SSH がデバイスで有効になっていることを示しています。

Step 2 SSH が無効になっているかどうかを確認するには、**show ip ssh** コマンドを使用します。

Example:

```
Device# show ip ssh
%SSH has not been enabled
```

上記の出力は、SSH がデバイスで有効になっていないことを示しています。

Step 3 SSH サーバー接続のステータスを確認するには、**show ssh** コマンドを使用します。

Example:

```
Device# show ssh
      Connection Version Encryption State Username
      0 1.5 3DES Session Started guest
```

この出力は、SSH を有効にしたときのデバイス上の SSH サーバー接続を示しています。

Step 4 **show ssh** コマンドを使用して、SSH サーバー接続が実行されているかどうかを確認します。

Example:

```
Device# show ssh
      %No SSH server connections running.
```

この出力は、デバイスで実行中の SSH サーバー接続がないことを示します。

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。