SSH および Telnet の設定

この章では、Cisco NX-OS デバイス上でセキュア シェル（SSH）プロトコルおよび Telnet を設定する手順について説明します。

この章は、次の項で構成されています。

- 機能情報の確認，1 ページ
- SSH および Telnet の概要，2 ページ
- SSH および Telnet のライセンス要件，4 ページ
- SSH および Telnet の前提条件，4 ページ
- SSH と Telnet の注意事項と制約事項，4 ページ
- SSH および Telnet のデフォルト設定，5 ページ
- SSH の設定，5 ページ
- Telnet の設定，20 ページ
- SSH および Telnet の設定の確認，23 ページ
- SSH の設定例，24 ページ
- SSH のパスワードが不要なファイル コピーの設定例，25 ページ
- SSH および Telnet に関する追加情報，26 ページ
- SSH および Telnet の機能の履歴，27 ページ

機能情報の確認

ご使用のソフトウェア リリースで、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。最新の警告および機能情報については、https://tools.cisco.com/bugsearch/ の Bug Search Tool およびご使用のソフトウェア リリースのリリース 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、および各機能がサポートされているリリースのリス

トについては、「新機能および変更された機能に関する情報」の章または以下の「機能の履歴」表を参照してください。

SSH および Telnet の概要

ここでは、SSH および Telnet について説明します。

SSH サーバ

SSH サーバを使用すると、SSH クライアントは Cisco NX-OS デバイスとの間で暗号化された安全な接続を確立できます。SSH は強化暗号化を使用して認証を行います。Cisco NX-OS ソフトウェアの SSH サーバは、市販の一般的な SSH クライアントと相互運用ができます。

SSH がサポートするユーザ認証メカニズムには、Remote Authentication Dial-In User Service（RADIUS）、TACACS+、LDAP、およびローカルに格納されたユーザ名とパスワードを使用した認証があります。

SSH クライアント

SSH クライアントは、SSH プロトコルで稼働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。Cisco NX-OS デバイスは、SSH クライアントを使用して、別の Cisco NX-OS デバイスまたは SSH サーバの稼働する他のデバイスとの間で暗号化された安全な接続を確立できます。この接続は、暗号化されたアウトバウンド接続を実現します。認証と暗号化により、SSH クライアントは、セキュリティ保護されていないネットワーク上でもセキュアな通信を実現できます。

Cisco NX-OS ソフトウェアの SSH クライアントは、市販の一般的な SSH クライアントと相互運用ができます。

SSH サーバ キー

SSH では、Cisco NX-OS デバイスと安全な通信を行うためにサーバ キーが必要です。SSH サーバ キーは、次の SSH オプションに使用できます。

- Rivest, Shamir, and Adelman（RSA）公開キー暗号化を使用した SSH バージョン 2
- Digital System Algrorithm（DSA）を使用した SSH バージョン 2

SSH サービスをイネーブルにする前に、適切なバージョンの SSH サーバ キー ペアを取得してください。使用中の SSH クライアント バージョンに応じて、SSH サーバ キー ペアを生成します。SSH サービスは、SSH バージョン 2 で使用する次の 2 種類のキー ペアを受け入れます。

- **dsa** オプションを使用すると、SSH バージョン 2 プロトコルに対応する DSA キーペアが生成されます。

- **rsa** オプションを使用すると、SSH バージョン 2 プロトコルに対応する RSA キーペアが生成されます。

デフォルトでは、Cisco NX-OS ソフトウェアは 1024 ビットの RSA キーを作成します。

SSH は、次の公開キー形式をサポートします。

- OpenSSH
- IETF SSH（SECSH）
- Privacy-Enhanced Mail（PEM）の公開キー証明書

⚠

注意 SSH キーをすべて削除すると、SSH サービスを開始できません。

デジタル証明書を使用した SSH 認証

Cisco NX-OS デバイスでの SSH 認証では、ホスト認証用に X.509 デジタル証明書をサポートしています。X.509 デジタル証明書は、メッセージの出所と整合性を保証するデータ項目です。これには安全な通信のための暗号化されたキーが含まれています。また、発信者のアイデンティティを証明するために信頼できる認証局（CA）によって署名されています。X.509 デジタル証明書のサポートにより、認証に DSA と RSA のいずれかのアルゴリズムを使用します。

証明書のインフラストラクチャでは、Secure Socket Layer（SSL）に対応し、セキュリティ インフラストラクチャによってクエリーまたは通知を通じて最初に返される証明書が使用されます。証明書が信頼できる CA のいずれかから発行されたものであれば、証明書の検証は成功です。

デバイスは、X.509 証明書を使用する SSH 認証用か、公開キー証明書を使用する SSH 認証用のいずれかに設定できますが、両方は不可能です。これらのいずれかが設定されているときに認証に失敗すると、パスワードの入力が求められます。

Telnet サーバ

Telnet プロトコルは、ホストとの TCP/IP 接続を確立します。Telnet を使用すると、あるサイトのユーザが別のサイトのログイン サーバと TCP 接続を確立し、キーストロークをデバイス間でやり取りできます。Telnet は、リモート デバイス アドレスとして IP アドレスまたはドメイン名のいずれかを受け入れます。

デフォルトでは、Telnet サーバは Cisco NX-OS デバイス上でディセーブルになっています。

SSH および Telnet のバーチャライゼーション サポート

SSH および Telnet の設定と操作は、仮想デバイス コンテキスト（VDC）に対してローカルです。VDC の詳細については、『*Cisco Nexus 7000 Series NX-OS Virtual Device Context Configuration Guide*』を参照してください。

SSH および Telnet のライセンス要件

次の表に、この機能のライセンス要件を示します。

製品	ライセンス要件
Cisco NX-OS	SSH および Telnet にはライセンスは必要ありません。ライセンスパッケージに含まれていない機能はすべて Cisco NX-OS システムイメージにバンドルされており、追加費用は一切発生しません。Cisco NX-OS ライセンス方式の詳細については、『*Cisco NX-OS Licensing Guide*』を参照してください。

SSH および Telnet の前提条件

SSH および Telnet の前提条件は次のとおりです。

- レイヤ 3 インターフェイス上に IP、mgmt 0 インターフェイス上にアウトバンド、またはイーサネット インターフェイス上にインバンドを設定していること

SSH と Telnet の注意事項と制約事項

SSH および Telnet に関する注意事項と制約事項は次のとおりです。

- Cisco NX-OS ソフトウェアは、SSH バージョン 2（SSHv2）だけをサポートしています。
- デバイスには、X.509 証明書を使用した SSH 認証か、または公開キー証明書を使用した SSH 認証のどちらかを設定できますが、その両方は設定できません。これらのいずれかが設定されているときに認証に失敗すると、パスワードの入力が求められます。
- Cisco NX-OS Release 5.1 から、SSH は FIPS モードで実行されます。
- SFTP サーバ機能では、通常の SFTP の **chown** および **chgrp** コマンドはサポートされません。
- SFTP サーバがイネーブルになっている場合は、admin ユーザだけが SFTP を使用してスイッチにアクセスできます。

（注） Cisco IOS の CLI に慣れている場合、この機能に対応する Cisco NX-OS コマンドは通常使用する Cisco IOS コマンドと異なる場合があるので注意してください。

関連トピック

FIPS の設定

SSH および Telnet のデフォルト設定

次の表に、SSH および Telnet パラメータのデフォルト設定を示します。

表 *1*：デフォルトの *SSH* および *Telnet* パラメータ

パラメータ	デフォルト
SSH サーバ	イネーブル
SSH サーバ キー	1024 ビットで生成された RSA キー
RSA キー生成ビット数	1024
Telnet サーバ	ディセーブル
Telnet ポート番号	23
SSH ログインの最大試行回数	3
SCP サーバ	ディセーブル
SFTP サーバ	ディセーブル

SSH の設定

ここでは、SSH の設定方法について説明します。

SSH サーバ キーの生成

セキュリティ要件に基づいてSSHサーバキーを生成できます。デフォルトのSSHサーバキーは、1024 ビットで生成される RSA キーです。

手順の概要

1. **configure terminal**
2. **no feature ssh**
3. **ssh key** {**dsa** [**force**] | **rsa** [*bits* [**force**]]}
4. **feature ssh**
5. **exit**
6. （任意） **show ssh key**
7. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	
	例：
	`switch# configure terminal`
	`switch(config)#`
		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 2	**no feature ssh** 例： `switch(config)# no feature ssh`	SSH をディセーブルにします。
ステップ 3	**ssh key** {**dsa** [**force**] | **rsa** [*bits* [**force**]]} 例： `switch(config)# ssh key rsa 2048`	SSH サーバ キーを生成します。 *bits* 引数には、RSA キーの生成に使用するビット数を指定します。Cisco NX-OS Release 5.1 以降では、範囲は 1024 ～ 2048 です。Cisco NX-OS Release 5.0 で、範囲は 768 ～ 2048 です。デフォルト値は 1024 です。 DSA キーのサイズを指定できません。これは常に 1024 ビットに設定されます。 既存のキーを置き換える場合は、キーワード **force** を使用します。
ステップ 4	**feature ssh** 例： `switch(config)# feature ssh`	SSH をイネーブルにします。

ステップ 5	**exit**
	
	例：
	`switch(config)# exit`
	`switch#`
		グローバル コンフィギュレーション モードを終了します。

	コマンドまたはアクション	目的
ステップ 6	**show ssh key** 例： `switch# show ssh key`	（任意） SSH サーバ キーを表示します。
ステップ 7	**copy running-config startup-config** 例： `switch# copy running-config startup-config`	（任意） 実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

ユーザ アカウント用 SSH 公開キーの指定

SSH 公開キーを設定すると、パスワードを要求されることなく、SSH クライアントを使用してログインできます。SSH 公開キーは、次のいずれかの形式で指定できます。

- OpenSSH 形式
- Internet Engineering Task Force（IETF）SECSH 形式
- Privacy Enhanced Mail（PEM）形式の公開キー証明書

IETF SECSH 形式による SSH 公開キーの指定

ユーザ アカウント用に IETF SECSH 形式で SSH 公開キーを指定できます。

はじめる前に

IETF SCHSH 形式の SSH 公開キーを作成します。

手順の概要

1. **copy***server-file***bootflash:***filename*
2. **configure terminal**
3. **username***username***sshkey file bootflash:***filename*
4. **exit**
5. （任意） **show user-account**
6. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**copy***server-file***bootflash:***filename* 例： `switch# copy tftp://10.10.1.1/secsh_file.pub bootflash:secsh_file.pub`	サーバから IETF SECSH 形式の SSH キーを含むファイルをダウンロードします。サーバは FTP、Secure Copy（SCP）、Secure FTP（SFTP）、または TFTP のいずれかを使用できます。
ステップ 2	**configure terminal** 例： `switch# configure terminal switch(config)#`	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	**username***username***sshkey file bootflash:***filename* 例： `switch(config)# username User1 sshkey file bootflash:secsh_file.pub`	IETF SECSH 形式の SSH 公開キーを設定します。
ステップ 4	**exit** 例： `switch(config)# exit switch#`	グローバル コンフィギュレーション モードを終了します。
ステップ 5	**show user-account** 例： `switch# show user-account`	（任意） ユーザ アカウントの設定を表示します。
ステップ 6	**copy running-config startup-config** 例： `switch# copy running-config startup-config`	（任意） 実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

OpenSSH 形式の SSH 公開キーの指定

ユーザ アカウントに OpenSSH 形式の SSH 公開キーを指定できます。

はじめる前に

OpenSSH 形式の SSH 公開キーを作成します。

手順の概要

1. **configure terminal**
2. **username** *username* **sshkey** *ssh-key*
3. **exit**
4. （任意） **show user-account**
5. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	
	例：
	`switch# configure terminal`
	`switch(config)#`
		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 2	**username** *username* **sshkey** *ssh-key* 例： `switch(config)# username User1 sshkey ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAIEAy19oF6QaZl9G+3f1XswK3OiW4H7YyUyuA50rv7gsEPjhOBYmsi6PAVKui1nIf/DQhum+lJNqJP/eLowb7ubO+lVKRXFY/G+lJNIQW3g9igG30c6k6+XVn+NjnI1B7ihvpVh7dLddMOXwOnXHYshXmSiH3UD/vKyziEh5S4Tplx8=`	OpenSSH 形式の SSH 公開キーを設定します。

ステップ 3	**exit**
	
	例：
	`switch(config)# exit`
	`switch#`
		グローバル コンフィギュレーション モードを終了します。

ステップ 4	**show user-account** 例： `switch# show user-account`	（任意） ユーザ アカウントの設定を表示します。
ステップ 5	**copy running-config startup-config** 例： `switch# copy running-config startup-config`	（任意） 実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

SSH ログイン試行の最大回数の設定

SSH ログイン試行の最大回数を設定できます。許可される試行の最大回数を超えると、セッションが切断されます。

（注） ログイン試行の合計回数には、公開キー認証、証明書ベースの認証、およびパスワードベースの認証を使用した試行が含まれます。イネーブルにされている場合は、公開キー認証が優先されます。証明書ベースとパスワード ベースの認証だけがイネーブルにされている場合は、証明書ベースの認証が優先されます。これらすべての方法で、ログイン試行の設定された数を超えると、認証失敗回数を超過したことを示すメッセージが表示されます。

手順の概要

1. **configure terminal**
2. **ssh login-attempts***number*
3. （任意） **show running-config security all**
4. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	
	例：
	`switch# configure terminal`
	`switch(config)#`
		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 2	**ssh login-attempts***number* 例： `switch(config)# ssh login-attempts 5`	ユーザが SSH セッションへのログインを試行できる最大回数を設定します。ログイン試行のデフォルトの最大回数は 3 です。値の範囲は 1 ～ 10 です。 （注） このコマンドの **no** 形式を使用すると、以前のログイン試行の値が削除され、ログイン試行の最大回数がデフォルト値の 3 に設定されます。
ステップ 3	**show running-config security all** 例： `switch(config)# show running-config security all`	（任意） SSH ログイン試行の設定された最大回数を表示します。
ステップ 4	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） （任意）実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーします。

SSH 接続のログイン猶予時間の設定

リモート デバイスから Cisco NX-OS デバイスへの SSH 接続に対しては、ログイン猶予時間を設定できます。この設定では、クライアントが認証を完了するまでの猶予時間を指定します。SSH セッションにログインするまでの時間が指定した猶予時間を超えるとセッションが切断され、再度ログインを試みる必要があります。

（注） リモート デバイスの SSH サーバをイネーブルにします。

手順の概要

1. **configure terminal**
2. **feature ssh**
3. **ssh login-gracetime***number*
4. （任意） **exit**
5. （任意） **show running-config security**
6. （任意） **show running-config security all**
7. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	
	例：
	`switch# configure terminal`
	`switch(config)#`
		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 2	**feature ssh**
	
	例：
	`switch# feature ssh`
	`switch(config)#`
		SSH をイネーブルにします。

ステップ 3	**ssh login-gracetime***number* 例： `switch(config)# ssh login-gracetime 120`	リモート デバイスから Cisco NX-OS デバイスへの SSH 接続のログイン猶予時間を秒単位で設定します。デフォルトのログイン猶予時間は 120 秒です。範囲は、1 ～ 2147483647 です。 （注） このコマンドの **no** 形式を使用すると設定したログイン猶予時間が削除され、デフォルト値の 120 秒にリセットされます。

	コマンドまたはアクション	目的
ステップ 4	**exit** 例： `switch(config)# exit`	（任意） グローバル コンフィギュレーション モードを終了します。
ステップ 5	**show running-config security** 例： `switch(config)# show running-config security`	（任意） 設定した SSH ログイン猶予時間が表示されます。
ステップ 6	**show running-config security all** 例： `switch(config)# show running-config security all`	（任意） デフォルトまたは設定した SSH ログイン猶予時間が表示されます。
ステップ 7	**copy running-config startup-config** 例： `switch(config)# copy running-config startup-config`	（任意） （任意）実行コンフィギュレーションをスタートアップ コンフィギュレーションにコピーします。

SSH セッションの開始

Cisco NX-OS デバイスから IPv4 または IPv6 を使用して SSH セッションを開始し、リモート デバイスと接続します。

はじめる前に

リモートデバイスのホスト名を取得し、必要なら、リモートデバイスのユーザ名も取得します。

リモート デバイスの SSH サーバをイネーブルにします。

手順の概要

1. **ssh** [*username*@]{*ipv4-address* | *hostname*} [**vrf***vrf-name*]
2. **ssh6** [*username*@]{*ipv6-address* | *hostname*} [**vrf** *vrf-name*]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**ssh** [*username*@]{*ipv4-address* | *hostname*} [**vrf***vrf-name*] 例： `switch# ssh 10.10.1.1`	IPv4 を使用してリモートデバイスとの SSH IPv4 セッションを作成します。デフォルトの VRF はデフォルト VRF です。
ステップ 2	**ssh6** [*username*@]{*ipv6-address* | *hostname*} [**vrf** *vrf-name*] 例： `switch# ssh6 HostA`	IPv6 を使用してリモートデバイスとの SSH IPv6 セッションを作成します。

ブート モードからの SSH セッションの開始

SSH セッションは、リモート デバイスに接続する Cisco NX-OS デバイスのブート モードから開始できます。

はじめる前に

リモートデバイスのホスト名を取得し、必要なら、リモートデバイスのユーザ名も取得します。

リモート デバイスの SSH サーバをイネーブルにします。

Cisco NX-OS デバイスがキックスタート イメージのみを使用してロードされていることを確認します。

手順の概要

1. **ssh** [*username*@]*hostname*
2. **exit**
3. **copy scp:**//[*username*@]*hostname*/*filepath directory*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**ssh** [*username*@]*hostname* 例： `switch(boot)# ssh user1@10.10.1.1`	リモート デバイスへの SSH セッションを、Cisco NX-OS デバイスのブート モードから作成します。デフォルト VRF が常に使用されます。

	コマンドまたはアクション	目的
ステップ 2	**exit** 例： `switch(boot)# exit`	ブート モードを終了します。
ステップ 3	**copy scp:**//[*username@*]*hostname*/*filepath directory* 例： `switch# copy scp://user1@10.10.1.1/users abc`	セキュア コピー プロトコル（SCP）を使用して、ファイルを Cisco NX-OS デバイスからリモート デバイスへコピーします。デフォルト VRF が常に使用されます。

SSH のパスワードが不要なファイル コピーの設定

Cisco NX-OS デバイスから Secure Copy（SCP）サーバまたは Secure FTP（SFTP）サーバに、パスワードなしでファイルをコピーすることができます。これを行うには、SSH による認証用の公開キーと秘密キーで構成される RSA または DSA のアイデンティティを作成する必要があります。

手順の概要

1. **configure terminal**
2. [**no**] **username** *username* **keypair generate** {**rsa** [*bits* [**force**]] | **dsa** [**force**]}
3. （任意） **show username** *username* **keypair**
4. **username** *username* **keypair export** {**bootflash:***filename* | **volatile:***filename*} {**rsa** | **dsa**} [**force**]
5. **username** *username* **keypair import** {**bootflash:***filename* | **volatile:***filename*} {**rsa** | **dsa**} [**force**]

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	
	例：
	`switch# configure terminal`
	`switch(config)#`
		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 2	[**no**] **username** *username* **keypair generate** {**rsa** [*bits* [**force**]] | **dsa** [**force**]} 例： `switch(config)# username user1 keypair generate rsa 2048 force`	SSH の公開キーと秘密キーを生成し、指定したユーザの Cisco NX-OS デバイスのホームディレクトリ（$HOME/.ssh）に格納します。Cisco NX-OS デバイスでは、これらのキーを使用してリモート マシンの SSH サーバと通信します。

	コマンドまたはアクション	目的
		bits 引数には、キーの生成に使用するビット数を指定します。Cisco NX-OS Release 5.1 以降では、範囲は 1024 ～ 2048 です。Cisco NX-OS Release 5.0 で、範囲は 768 ～ 2048 です。デフォルト値は 1024 です。 既存のキーを置き換える場合は、キーワード **force** を使用します。**force** キーワードを省略した場合、SSH キーがすでに存在していれば、SSH キーは生成されません。
ステップ 3	**show username** *username* **keypair** 例： `switch(config)# show username user1 keypair`	（任意） 指定したユーザの公開キーを表示します。 （注） セキュリティ上の理由から、このコマンドで秘密キーは表示されません。
ステップ 4	**username** *username* **keypair export** {**bootflash:***filename* | **volatile:***filename*} {**rsa** | **dsa**} [**force**] 例： `switch(config)# username user1 keypair export bootflash:key_rsa rsa`	Cisco NX-OS デバイスのホーム ディレクトリから、指定したブートフラッシュ ディレクトリまたは一時ディレクトリに、公開キーと秘密キーをエクスポートします。 既存のキーを置き換える場合は、キーワード **force** を使用します。**force** キーワードを省略した場合、SSH キーがすでに存在していれば、SSH キーはエクスポートされません。 生成したキー ペアをエクスポートするとき、秘密キーを暗号化するパスフレーズを入力するように求められます。秘密キーは、指定したファイルとしてエクスポートされ、公開キーは、同じファイル名に .pub 拡張子を付けてエクスポートされます。これで、このキー ペアを任意の Cisco NX-OS デバイスにコピーし、SCP または SFTP を使用してサーバのホーム ディレクトリに公開キー ファイル（*.pub）をコピーできるようになります。 （注） セキュリティ上の理由から、このコマンドはグローバル コンフィギュレーション モードでしか実行できません。
ステップ 5	**username** *username* **keypair import** {**bootflash:***filename* | **volatile:***filename*} {**rsa** | **dsa**} [**force**] 例： `switch(config)# username user1 keypair import bootflash:key_rsa rsa`	指定したブートフラッシュ ディレクトリまたは一時ディレクトリから、Cisco NX-OS デバイスのホーム ディレクトリに、エクスポートした公開キーと秘密キーをインポートします。 既存のキーを置き換える場合は、キーワード **force** を使用します。**force** キーワードを省略した場合、SSH キーがすでに存在していれば、SSH キーはインポートされません。 生成したキー ペアをインポートするとき、秘密キーを復号化するパスフレーズを入力するように求められます。秘密キーは指定したファイルとしてインポートされ、公開キーは同じファイル名に .pub 拡張子を付けてインポートされます。 （注） セキュリティ上の理由から、このコマンドはグローバル コンフィギュレーション モードでしか実行できません。

	コマンドまたはアクション	目的
		（注） パスワードなしでサーバにアクセスできるのは、サーバでキーが設定されているユーザのみです。

次の作業

SCP サーバまたは SFTP サーバで、次のコマンドを使用して、*.pub ファイル（たとえば、key_rsa.pub）に格納された公開キーを authorized_keys ファイルに追加します。

$ cat key_rsa.pub >> $HOME/.ssh/ authorized_keys

これで、標準の SSH コマンドおよび SCP コマンドを使用してパスワードを指定しなくても、Cisco NX-OS デバイスからサーバにファイルをコピーできます。

SCP サーバと SFTP サーバの設定

リモート デバイスとの間でファイルをコピーできるように、Cisco NX-OS デバイスで SCP サーバまたは SFTP サーバを設定できます。SCP サーバまたは SFTP サーバをイネーブルにした後、Cisco NX-OS デバイスとの間でファイルをコピーするために、リモート デバイスで SCP または SFTP コマンドを実行できます。

（注） arcfour および blowfish cipher オプションは SCP サーバではサポートされません。

手順の概要

1. **configure terminal**
2. [**no**] **feature scp-server**
3. [**no**] **feature sftp-server**
4. **exit**
5. （任意） **show running-config security**
6. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	
	例：
	`switch# configure terminal`
	`switch(config)#`
		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。

	コマンドまたはアクション	目的
ステップ 2	**[no] feature scp-server** 例： `switch(config)# feature scp-server`	Cisco NX-OS デバイスで SCP サーバをイネーブルまたはディセーブルにします。
ステップ 3	**[no] feature sftp-server** 例： `switch(config)# feature sftp-server`	Cisco NX-OS デバイスで SFTP サーバをイネーブルまたはディセーブルにします。

ステップ 4	**exit**
	
	例：
	`switch(config)# exit`
	`switch#`
		グローバル コンフィギュレーション モードを終了します。

ステップ 5	**show running-config security** 例： `switch# show running-config security`	（任意） SCP サーバと SFTP サーバの設定ステータスを表示します。
ステップ 6	**copy running-config startup-config** 例： `switch# copy running-config startup-config`	（任意） 実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

SSH ホストのクリア

サーバから SCP または SFTP を使用してファイルをダウンロードする場合、またはこのデバイスからリモート ホストに SSH セッションを開始する場合には、そのサーバと信頼できる SSH 関係が確立されます。ユーザアカウントの、信頼できる SSH サーバのリストはクリアすることができます。

手順の概要

1. **clear ssh hosts**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**clear ssh hosts** 例： `switch# clear ssh hosts`	SSH ホスト セッションおよび既知のホスト ファイルをクリアします。

SSH サーバのディセーブル化

デフォルトでは、SSH サーバは Cisco NX-OS デバイス上でイネーブルになっています。SSH サーバをディセーブルにすると、SSH でスイッチにアクセスすることを防止できます。

手順の概要

1. **configure terminal**
2. **no feature ssh**
3. **exit**
4. （任意） **show ssh server**
5. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	
	例：
	`switch# configure terminal`
	`switch(config)#`
		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 2	**no feature ssh** 例： `switch(config)# no feature ssh`	SSH をディセーブルにします。

ステップ 3	**exit**
	
	例：
	`switch(config)# exit`
	`switch#`
		グローバルコンフィギュレーションモードを終了します。

ステップ 4	**show ssh server** 例： `switch# show ssh server`	（任意） SSH サーバの設定を表示します。
ステップ 5	**copy running-config startup-config** 例： `switch# copy running-config startup-config`	（任意） 実行コンフィギュレーションを、スタートアップコンフィギュレーションにコピーします。

SSH サーバ キーの削除

SSH サーバをディセーブルにした後、Cisco NX-OS デバイス上の SSH サーバ キーを削除できます。

（注） SSH を再度イネーブルにするには、まず、SSH サーバ キーを生成する必要があります。

手順の概要

1. **configure terminal**
2. **no feature ssh**
3. **no ssh key** [**dsa** | **rsa**]
4. **exit**
5. （任意） **show ssh key**
6. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	
	例：
	`switch# configure terminal`
	`switch(config)#`
		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 2	**no feature ssh** 例： `switch(config)# no feature ssh`	SSH をディセーブルにします。
ステップ 3	**no ssh key** [**dsa** | **rsa**] 例： `switch(config)# no ssh key rsa`	SSH サーバ キーを削除します。 デフォルトでは、すべての SSH キーが削除されます。

ステップ 4	**exit**
	
	例：
	`switch(config)# exit`
	`switch#`
		グローバル コンフィギュレーション モードを終了します。

ステップ 5	**show ssh key** 例： `switch# show ssh key`	（任意） SSH サーバ キーの設定を表示します。

	コマンドまたはアクション	目的
ステップ 6	**copy running-config startup-config** 例： `switch# copy running-config startup-config`	（任意） 実行コンフィギュレーションを、スタートアップコンフィギュレーションにコピーします。

関連トピック

SSH サーバ キーの生成，（5 ページ）

SSH セッションのクリア

Cisco NX-OS デバイスから SSH セッションを クリアできます。

手順の概要

1. **show users**
2. **clear line***vty-line*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**show users** 例： `switch# show users`	ユーザ セッション情報を表示します。
ステップ 2	**clear line***vty-line* 例： `switch(config)# clear line pts/12`	ユーザ SSH セッションをクリアします。

Telnet の設定

ここでは、Cisco NX-OS デバイスで Telnet を設定する手順を説明します。

Telnet サーバのディセーブル化

Telnet サーバを Cisco NX-OS デバイス上でイネーブルにできます。デフォルトでは、Telnet はディセーブルです。

手順の概要

1. **configure terminal**
2. **feature telnet**
3. **exit**
4. （任意） **show telnet server**
5. （任意） **copy running-config startup-config**

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	**configure terminal**
	
	例：
	`switch# configure terminal`
	`switch(config)#`
		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 2	**feature telnet** 例： `switch(config)# feature telnet`	Telnet サーバをイネーブルにします。デフォルトではディセーブルになっています。

ステップ 3	**exit**
	
	例：
	`switch(config)# exit`
	`switch#`
		グローバル コンフィギュレーション モードを終了します。

ステップ 4	**show telnet server** 例： `switch# show telnet server`	（任意） Telnet サーバの設定を表示します。
ステップ 5	**copy running-config startup-config** 例： `switch# copy running-config startup-config`	（任意） 実行コンフィギュレーションを、スタートアップ コンフィギュレーションにコピーします。

リモート デバイスとの Telnet セッションの開始

Cisco NX-OS デバイスから SSH セッションを開始して、リモート デバイスと接続できます。IPv4 または IPv6 のいずれかを使用して Telnet セッションを開始できます。

はじめる前に

リモート デバイスのホスト名または IP アドレスと、必要な場合はリモート デバイスのユーザ名を取得します。

Cisco NX-OS デバイス上で Telnet サーバをイネーブルにします。

リモート デバイス上で Telnet サーバをイネーブルにします。

手順の概要

1. **telnet** {*ipv4-address* | *host-name*} [*port-number*] [**vrf***vrf-name*]
2. **telnet6** {*ipv6-address* | *host-name*} [*port-number*] [**vrf***vrf-name*]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**telnet** {*ipv4-address* | *host-name*} [*port-number*] [**vrf***vrf-name*] 例： `switch# telnet 10.10.1.1`	IPv4 を使用してリモート デバイスとの Telnet セッションを開始します。デフォルトのポート番号は 23 です。有効な範囲は 1 ～ 65535 です。デフォルトの VRF はデフォルト VRF です。
ステップ 2	**telnet6** {*ipv6-address* | *host-name*} [*port-number*] [**vrf***vrf-name*] 例： `switch# telnet6 2001:0DB8::ABCD:1 vrf management`	IPv6 を使用してリモート デバイスとの Telnet セッションを開始します。デフォルトのポート番号は 23 です。有効な範囲は 1 ～ 65535 です。デフォルトの VRF はデフォルト VRF です。

関連トピック

Telnet サーバのディセーブル化, （21 ページ）

Telnet セッションのクリア

Telnet セッションを Cisco NX-OS デバイスからクリアできます。

はじめる前に

Cisco NX-OS デバイス上で Telnet サーバをイネーブルにします。

手順の概要

1. **show users**
2. **clear line***vty-line*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**show users** 例： `switch# show users`	ユーザ セッション情報を表示します。
ステップ 2	**clear line***vty-line* 例： `switch(config)# clear line pts/12`	ユーザ Telnet セッションをクリアします。

SSH および Telnet の設定の確認

SSH および Telnet の設定情報を表示するには、次のいずれかの作業を行います。

コマンド	目的
show ssh key [**dsa** | **rsa**]	SSH サーバ キー ペアの情報を表示します。
show running-config security [**all**]	実行コンフィギュレーション内の SSH とユーザ アカウントの設定を表示します。キーワード **all** を指定すると、SSH およびユーザ アカウントのデフォルト値が表示されます。
show ssh server	SSH サーバの設定を表示します。
show telnet server	Telnet サーバの設定を表示します。
show username *username* **keypair**	指定したユーザの公開キーを表示します。

これらのコマンド出力のフィールドの詳細については、『*Cisco Nexus 7000 Series NX-OS Security Command Reference*』を参照してください。

SSH の設定例

次の例は、OpenSSH キーを使用して SSH を設定する方法を示しています。

ステップ 1 SSH サーバをディセーブルにします。

例：
```
switch# configure terminal
switch(config)# no feature ssh
```

ステップ 2 SSH サーバ キーを生成します。

例：

```
switch(config)# ssh key rsa
generating rsa key(1024 bits)......
generated rsa key
```

ステップ 3 SSH サーバをイネーブルにします。

例：
```
switch(config)# feature ssh
```

ステップ 4 SSH サーバ キーを表示します。

例：
```
switch(config)# show ssh key
rsa Keys generated:Sat Sep 29 00:10:39 2007

ssh-rsa AAAAB3NzaC1yc2EAAAABIwAAAIEAvWhEBsF55oaPHNDBnpXOTw6+/OdHoLJZKr
+MZm99n2U0ChzZG4svRWmHuJY4PeDWl0e5yE3g3EO3pjDDmt923siNiv5aSga60K36lr39
HmXL6VgpRVn1XQFiBwn4na+H1d3Q0hDt+uWEA0tka2uOtXlDhliEmn4HVXOjGhFhoNE=

bitcount:1024
fingerprint:
51:6d:de:1c:c3:29:50:88:df:cc:95:f0:15:5d:9a:df
*************************************
could not retrieve dsa key information
*************************************
```

ステップ 5 OpenSSH 形式の SSH 公開キーを指定します。

例：
```
switch(config)# username User1 sshkey ssh-rsa
AAAAB3NzaC1yc2EAAAABIwAAAIEAy19oF6QaZl9G+3f1XswK3OiW4H7YyUyuA50r
v7gsEPjhOBYmsi6PAVKui1nIf/DQhum+lJNqJP/eLowb7ubO+lVKRXFY/G+lJNIQ
W3g9igG30c6k6+XVn+NjnI1B7ihvpVh7dLddMOXwOnXHYshXmSiH3UD/vKyziEh5
4Tplx8=
```

ステップ 6 設定を保存します。

例：
```
switch(config)# copy running-config startup-config
```

SSH のパスワードが不要なファイル コピーの設定例

次に、Cisco NX-OS デバイスから Secure Copy（SCP）サーバまたは Secure FTP（SFTP）サーバに、パスワードなしでファイルをコピーする例を示します。

ステップ 1 SSH の公開キーと秘密キーを生成し、指定したユーザの Cisco NX-OS デバイスのホーム ディレクトリに格納します。

例：
```
switch# configure terminal
switch(config)# username admin keypair generate rsa
generating rsa key(1024 bits)......
generated rsa key
```

ステップ 2 指定したユーザの公開キーを表示します。

例：
```
switch(config)# show username admin keypair

**************************************

rsa Keys generated: Thu Jul  9 11:10:29 2009

ssh-rsa
AAAAB3NzaC1yc2EAAAABIwAAAIEAxWmjJT+oQhIcvnrMbx2BmD0P8boZElTfJ
Fx9fexWp6rOiztlwODtehnjadWc6A+DE2DvYNvqsrU9TBypYDPQkR/+Y6cKubyFW
VxSBG/NHztQc3+QC1zdkIxGNJbEHyFoajzNEO8LLOVFIMCZ2Td7gxUGRZc+fbq
S33GZsCAX6v0=

bitcount:262144
fingerprint:
8d:44:ee:6c:ca:0b:44:95:36:d0:7d:f2:b5:78:74:7d
**************************************

could not retrieve dsa key information
**************************************
```

ステップ 3 Cisco NX-OS デバイスのホームディレクトリから、指定したブートフラッシュディレクトリに、公開キーと秘密キーをエクスポートします。

例：
```
switch(config)# username admin keypair export bootflash:key_rsa rsa
Enter Passphrase:
switch(config)# dir
.
```

```
.
.
        951     Jul 09 11:13:59 2009  key_rsa
        221     Jul 09 11:14:00 2009  key_rsa.pub
.
.
```

ステップ 4 **copy scp** または **copy sftp** コマンドを使用して、別の Cisco NX-OS デバイスにこれら 2 つのファイルをコピーした後、Cisco NX-OS デバイスのホーム ディレクトリにインポートします。

例：

```
switch(config)# username admin keypair import bootflash:key_rsa rsa
Enter Passphrase:
switch(config)# show username admin keypair
**************************************

rsa Keys generated: Thu Jul  9 11:10:29 2009

ssh-rsa
AAAAB3NzaC1yc2EAAAABIwAAAIEAxWmjJT+oQhIcvnrMbx2BmD0P8boZElTfJ
Fx9fexWp6rOiztlwODtehnjadWc6A+DE2DvYNvqsrU9TBypYDPQkR/+Y6cKubyFW
VxSBG/NHztQc3+QC1zdkIxGNJbEHyFoajzNEO8LLOVFIMCZ2Td7gxUGRZc+fbq
S33GZsCAX6v0=

bitcount:262144
fingerprint:
8d:44:ee:6c:ca:0b:44:95:36:d0:7d:f2:b5:78:74:7d
**************************************

could not retrieve dsa key information
**************************************
switch(config)#
```

ステップ 5 SCP サーバまたは SFTP サーバで、key_rsa.pub に格納されている公開キーを authorized_keys ファイルに追加します。

例：

```
$ cat key_rsa.pub >> $HOME/.ssh/ authorized_keys
```

これで、標準の SSH コマンドおよび SCP コマンドを使用してパスワードを指定しなくても、Cisco NX-OS デバイスからサーバにファイルをコピーできます。

ステップ 6 （任意） DSA キーについてこの手順を繰り返します。

SSH および Telnet に関する追加情報

ここでは、SSH および Telnet の実装に関する追加情報について説明します。

関連資料

関連項目	マニュアル タイトル
Cisco NX-OS のライセンス	『*Cisco NX-OS Licensing Guide*』
コマンド リファレンス	『*Cisco Nexus 7000 Series NX-OS Security Command Reference*』
VRF コンフィギュレーション	『*Cisco Nexus 7000 Series NX-OS Unicast Routing Configuration Guide*』

標準

標準	タイトル
この機能でサポートされる新規の標準または変更された標準はありません。また、既存の標準のサポートは変更されていません。	—

MIB

MIB	MIB のリンク
• CISCO-SECURE-SHELL-MIB	MIB を検索およびダウンロードするには、次の URL にアクセスしてください。 http://www.cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

SSH および Telnet の機能の履歴

次の表に、これらの機能のリリースの履歴を示します。

表 *2*：*SSH* および *Telnet* の機能の履歴

機能名	リリース	機能情報
SSH および Telnet	7.2(0)D1(1)	SSH ログインの最大試行回数の設定のサポートを追加しました。

機能名	リリース	機能情報
SSH および Telnet	6.0(1)	Release 5.2 以降、変更はありません。
SSH および Telnet	5.2(1)	Release 5.1 以降、変更はありません。
SCP および SFTP サーバ	5.1(1)	リモート デバイスとの間でのファイルのコピーをサポートするために、Cisco NX-OS デバイスで SCP サーバおよび SFTP サーバを設定する機能が追加されました。
SSH	5.1(1)	SSH は FIPS モードで実行されます。
SSH	5.1(1)	最小の RSA キー サイズが 768 ビットから 1024 ビットに変更されました。
SSH	5.0(2)	SSH ログインの最大試行回数の設定のサポートを追加しました。
SSH	5.0(2)	リモート デバイスに接続するために、Cisco NX-OS デバイスのブート モードから SSH セッションを開始する機能が追加されました。
SSH	5.0(2)	パスワードなしで Cisco NX-OS デバイスから SCP または SFTP サーバにファイルをコピーする機能が追加されました。
PKI	5.0(2)	リモートの cert-store のサポートを追加しました。