



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。

- [SSH および Telnet について, on page 1](#)
- [SSH および Telnet の前提条件, on page 3](#)
- [SSH と Telnet の注意事項と制約事項 \(3 ページ\)](#)
- [SSH および Telnet のデフォルト設定, on page 5](#)
- [SSH の設定 , on page 5](#)
- [Telnet の設定, on page 26](#)
- [SSH および Telnet の設定の確認, on page 28](#)
- [SSH の設定例, on page 29](#)
- [SSH のパスワードが不要なファイル コピーの設定例, on page 30](#)
- [X.509v3 証明書ベースの SSH 認証の設定例 \(32 ページ\)](#)
- [SSH および Telnet に関する追加情報, on page 33](#)

SSH および Telnet について

ここでは、SSH および Telnet について説明します。

SSH サーバー

SSH サーバを使用すると、SSH クライアントは Cisco NX-OS デバイスとの間でセキュアな暗号化された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、無償あるいは商用の SSH サーバと連係して動作します。

SSH サーバキー

SSH では、Cisco NX-OS とのセキュア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 Algorithm (DSA) を使用した SSH バージョン 2
- 楕円曲線デジタル署名アルゴリズム (ECDSA) 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 サービスでは、SSH バージョン 2 に対応する以下の 2 通りのキーペアを使用できます。

- **dsa** オプションでは、SSH バージョン 2 プロトコル用の DSA キーペアを作成します。
- **rsa** オプションでは、SSH バージョン 2 プロトコル用の RSA キーペアを作成します。
- **ecdsa** オプションでは、SSH バージョン 2 プロトコル用の ECDSA キーペアを作成します。

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

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



Caution 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 のいずれかで設定されており、無効にされたり期限が切れたりしていなければ、証明書の検証は成功します。

X.509 証明書を使用する SSH 認証用にデバイスを設定できます。認証に失敗した場合は、パスワードの入力が求められます。

X.509v3 証明書（RFC 6187）を使用する SSH 認証を設定できます。X.509v3 証明書ベースの SSH 認証では、スマートカードと組み合わせた証明書を使用して、シスコ デバイスへのアクセスの 2 要素認証を有効にします。SSH クライアントは、シスコパートナーの Pragma Systems によって提供されます。

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 サーバが 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 の前提条件

レイヤ 3 インターフェイス上で IP、mgmt 0 インターフェイス上でアウトバンド、またはイーサネット インターフェイス上でインバンドを設定していることを確認します。

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。
- Cisco NX-OS は、リモート TACACS 認証をサポートしていません。
- **no feature ssh feature** コマンドを使用すると、ポート 22 はディセーブルになりません。ポート 22 は常にオープンで、すべての着信外部接続を拒否する拒否ルールがプッシュされます。

- Poodle の脆弱性により、SSLv3 はサポートされなくなりました。
- IPSG は、次のものではサポートされません。
 - Cisco Nexus 9372PX、9372TX、および 9332PQ スイッチの最後の 6 個の 40 Gb 物理ポート
 - Cisco Nexus 9396PX、9396TX、および 93128TX スイッチのすべての 40G 物理ポート
- X.509 証明書を使用する SSH 認証用にデバイスを設定できます。認証に失敗した場合は、パスワードの入力が求められます。
- SFTP サーバ機能では、通常の SFTP の **chown** および **chgrp** コマンドを発行します。
- SFTP サーバが有効になっている場合は、**admin** ユーザだけが SFTP を使用してデバイスにアクセスできます。
- SSH パスワードレスファイルコピーを目的として AAA プロトコル (RADIUS や TACACS+ など) を介してリモート認証されたユーザアカウントにインポートされた SSH 公開キーと秘密キーは、同じ名前のローカルユーザアカウントでない限り、Nexus デバイスがリロードされると保持されません。リモートユーザアカウントは、SSH キーがインポートされる前にデバイスで設定されます。
- SSH のタイムアウト時間は、**tac-pac** の生成時間よりも長くする必要があります。そうでないと、VSH ログに **%VSHD-2-VSHD_SYSLOG_EOL_ERR** エラーが記録されることがあります。理想的には、**tac-pac** または **showtech** を収集する前に 0 (無限) に設定します。



(注) Cisco IOS の CLI に慣れている場合、この機能の Cisco NX-OS コマンドは従来の Cisco IOS コマンドと異なる点があるため注意が必要です。

- Cisco NX-OS Release 10.2(2)F 以降、新しい非同期化 CLI が導入され、SNMP とセキュリティコンポーネントの間のユーザー同期を無効にするオプションを提供します。詳細については、システム管理構成ガイドの **SNMP** の構成の章を参照してください。
リリース 7.0(3)I7(1) から現在のリリースまでのさまざまな機能をサポートする Cisco Nexus 9000 スイッチの詳細については、[Nexus スイッチプラットフォーム サポートマトリックス](#) を参照してください。
- 非同期 CLI が有効になっている場合、リモートユーザーは SNMP データベースに同期されません。
- DCNM (リリース 12.0.1.a 以降 Nexus Dashboard Fabric Controller と呼ばれる) を使用したセキュリティユーザーには、非同期 CLI が有効でないとき、対応する SNMPv3 プロファイルが存在しません。同期が無効になっている場合、セキュリティコンポーネントで作成されたユーザーはスイッチにログインできますが、コントローラはスイッチを検出しません。コントローラは、セキュリティユーザー用に作成された SNMP 構成を使用してスイッチを検出するためです。さらに、SNMP は、userDB の非同期状態のため、作成されたセキュリティユーザーを認識しないので、スイッチを検出できません。したがって、コント

ローラによってスイッチが検出されるようにするには、SNMPユーザーを明示的に作成する必要があります。DCNM 機能とともに非同期 CLI を使用することはお勧めしません。詳細については、*Cisco Nexus 9000 NX-OS セキュリティ構成ガイド*を参照してください。

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。

Table 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。

Procedure

	Command or Action	Purpose
ステップ 1	configure terminal Example: 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。

	Command or Action	Purpose
ステップ 2	no feature ssh Example: <pre>switch(config)# no feature ssh</pre>	SSH を無効にします。
ステップ 3	ssh key {dsa [force] rsa [bits[force]] ecdsa [bits [force]]} Example: <pre>switch(config)# ssh key rsa 2048</pre>	<p>SSH サーバ キーを生成します。</p> <p><i>bits</i> 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。有効な範囲は 768 ~ 2048 です。デフォルト値は 1024 です。</p> <p>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。</p> <p>既存のキーを置き換える場合は、force キーワードを使用します。</p> <p>Note <code>ssh key dsa</code> を設定する場合は、次の追加設定を行う必要があります：<code>ssh keytypes all</code> および <code>ssh kexalgs all</code></p>
ステップ 4	ssh rekey max-data max-data max-time max-time Example: <pre>switch(config)# ssh rekey max-data 1K max-time 1M</pre>	キー再生成パラメータを設定します。
ステップ 5	feature ssh Example: <pre>switch(config)# feature ssh</pre>	SSH を有効にします。
ステップ 6	exit Example: <pre>switch(config)# exit switch#</pre>	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 7	(Optional) show ssh key [dsa rsa ecdsa] [md5] Example: <pre>switch# show ssh key</pre>	<p>SSH サーバ キーを表示します。</p> <p>このコマンドは、デフォルトで SHA256 形式でフィンガープリントを表示します。SHA256 は、以前のデフォルトの MD5 形式よりも安全です。ただし、フィンガープリントを MD5 形式で表示する必要がある場合の下位互換性のために、md5 オプションが追加されています。</p>

	Command or Action	Purpose
ステップ 8	show run security all	
ステップ 9	(Optional) copy running-config startup-config Example: switch# copy running-config startup-config	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

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。

Before you begin

IETF SECSH 形式の SSH 公開キーを作成します。

Procedure

	Command or Action	Purpose
ステップ 1	copy server-file bootflash:filename Example: 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 Example: 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 Example: 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。

OpenSSH 形式の SSH 公開キーの指定

	Command or Action	Purpose
ステップ 4	exit Example: 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	(Optional) show user-account Example: switch# show user-account	ユーザアカウントの設定を表示します。
ステップ 6	(Optional) copy running-config startup-config Example: switch# copy running-config startup-config	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。

Before you begin

OpenSSH 形式の SSH 公開キーを作成します。

Procedure

	Command or Action	Purpose
ステップ 1	configure terminal Example: 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 Example: switch(config)# username User1 sshkey ssh-rsa AAAENaClyG2WWEIWAIPz19FQz19G3FDsK0V47MjyA50v7gEP h0Bnsi4ZAKiInIF/0Jm+LNgP/6o7JtG4IKRF/G4LNQ89ig3066 Xm+njn1LB7ihvMh7clldMOwOXHshXtSiH3UD/vKzyiEh59Tpk2=	OpenSSH 形式の SSH 公開キーを設定します。
ステップ 3	exit Example: 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。

	Command or Action	Purpose
ステップ 4	(Optional) show user-account Example: switch# show user-account	ユーザアカウントの設定を表示します。
ステップ 5	(Optional) copy running-config startup-config Example: switch# copy running-config startup-config	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。



Note 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。

Procedure

	Command or Action	Purpose
ステップ 1	configure terminal Example: 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 Example: 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。 Note 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。

	Command or Action	Purpose
ステップ 3	(Optional) show running-config security all Example: switch(config)# show running-config security all	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	(Optional) copy running-config startup-config Example: switch(config)# 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 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。

Before you begin

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 サーバを有効にします。

Procedure

	Command or Action	Purpose
ステップ 1	ssh [username@]{ipv4-address hostname} [vrf vrf-name] Example: 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] Example: 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。

Before you begin

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 サーバを有効にします。

Procedure

	Command or Action	Purpose
ステップ 1	ssh <i>[username@]hostname</i> Example: 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。
ステップ 2	exit Example: 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 Example: 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。

Procedure

	Command or Action	Purpose
ステップ 1	configure terminal Example: 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]} Example: 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

	Command or Action	Purpose
		<p>使用してリモート マシンの SSH サーバと通信します。</p> <p><i>bits</i> 引数には、キーの生成に使用するビット数を指定します。有効な範囲は 768 ~ 2048 です。デフォルト値は 1024 です。</p> <p>既存のキーを置き換える場合は、force キーワードを使用します。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。</p>
ステップ 3	<p>(Optional) show username <i>username</i> keypair</p> <p>Example:</p> <pre>switch(config)# show username user1 keypair</pre>	<p>指定したユーザの公開キーを表示します。</p> <p>Note 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。</p>
ステップ 4	<p>Required: username <i>username</i> keypair export {bootflash:<i>filename</i> volatile:<i>filename</i>} {rsa dsa} [force]</p> <p>Example:</p> <pre>switch(config)# username user1 keypair export bootflash:key_rsa rsa</pre>	<p>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。</p> <p>既存のキーを置き換える場合は、force キーワードを使用します。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。</p> <p>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。</p>

	Command or Action	Purpose
		<p>Note 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。</p>
ステップ 5	<p>Required: username <i>username</i> keypair import {bootflash:<i>filename</i> volatile:<i>filename</i>} {rsa dsa} [force]</p> <p>Example:</p> <pre>switch(config)# username user1 keypair import bootflash:key_rsa rsa</pre>	<p>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。</p> <p>既存のキーを置き換える場合は、force キーワードを使用します。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。</p> <p>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ワード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。</p> <p>Note 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。</p> <p>Note 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。</p>

What to do next

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。



Note 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。

Procedure

	Command or Action	Purpose
ステップ 1	configure terminal Example: switch# configure terminal switch(config)#	グローバル設定モードを開始します。
ステップ 2	[no] feature scp-server Example: switch(config)# feature scp-server	Cisco NX-OS デバイス上で 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	Required: [no] feature sftp-server Example: switch(config)# feature sftp-server	Cisco NX-OS デバイス上で 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	Required: exit Example: 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	(Optional) show running-config security Example: switch# show running-config security	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	(Optional) copy running-config startup-config Example: switch# copy running-config startup-config	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。

X.509v3 証明書ベースの SSH 認証の設定

X.509v3 証明書を使用する SSH 認証を設定できます。Cisco NX-OS は、リモート TACACS 認証をサポートしていません。

始める前に

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

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例 : <pre>switch# configure terminal switch(config)#</pre>	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 <i>user-id</i> [password [0 5] <i>password</i>] 例 : <pre>switch(config)# username jsmith password 4Ty18Rnt</pre>	<p>ユーザ アカウントを設定します。 <i>user-id</i> 引数は、最大 28 文字の英数字で、大文字と小文字が区別されます。指定できる文字は、A ~ Z の英大文字、a ~ z の英小文字、0 ~ 9 の数字、ハイフン (-)、ピリオド (.)、アンダースコア (_)、プラス符号 (+)、および等号 (=) です。アットマーク (@) はリモートユーザ名では使用できますが、ローカルユーザ名では使用できません。</p> <p>ユーザ名の先頭は英数字で始まる必要があります。</p> <p>デフォルトパスワードは定義されていません。オプションの 0 は、パスワードがクリアテキストであり、5 はパスワードが暗号化されていることを意味します。デフォルトは 0 (クリア テキスト) です。</p> <p>(注) パスワードを指定しなかった場合、ユーザは Cisco NX-OS デバイスにログインできません。</p>

	コマンドまたはアクション	目的
		<p>(注) 暗号化パスワードオプションを使用してユーザアカウントを作成する場合、対応する SNMP ユーザは作成されません。</p> <p>(注) 非同期 CLI が有効になっている場合、ユーザーアカウントを作成しても、対応する SNMP ユーザーは作成されません。</p>
ステップ 3	<p>username <i>user-id</i> ssh-cert-dn <i>dn-name</i> {dsa rsa}</p> <p>例 :</p> <pre>switch(config)# username jsmith ssh-cert-dn "/O = ABCcompany, OU = ABC1, emailAddress = jsmith@ABCcompany.com, L = Metropolis, ST = New York, C = US, CN = jsmith" rsa</pre>	既存のユーザアカウント認証に使用する SSH X.509 証明書の識別名と DSA アルゴリズムを指定します。識別名は最大 512 文字で、例に示す形式に従う必要があります。電子メールアドレスと状態がそれぞれ emailAddress と ST に設定されていることを確認します。
ステップ 4	<p>[no] crypto ca trustpoint <i>trustpoint</i></p> <p>例 :</p> <pre>switch(config)# crypto ca trustpoint winca switch(config-trustpoint)#</pre>	<p>トラストポイントを設定します。</p> <p>(注) このコマンドの no 形式を使用してトラストポイントを削除する前に、まず delete crl および delete ca-certificate コマンドを使用して、CRL および CA 証明書を削除する必要があります。</p>
ステップ 5	<p>crypto ca authenticate <i>trustpoint</i></p> <p>例 :</p> <pre>switch(config-trustpoint)# crypto ca authenticate winca</pre>	<p>トラストポイントの CA 証明書を設定します。</p> <p>(注) CA 証明書を削除するには、トラストポイントコンフィギュレーションモードで delete ca-certificate コマンドを入力します。</p>
ステップ 6	<p>(任意) crypto ca crl request <i>trustpoint</i> bootflash:static-crl.crl</p> <p>例 :</p>	この項はオプションですが、強く推奨されます。トラストポイントの証明書失効リスト (CRL) を設定します。CRL ファイルは、トラストポイントによっ

	コマンドまたはアクション	目的
	<pre>switch(config-trustpoint)# crypto ca crl request winca bootflash:crllist.crl</pre>	<p>て失効した証明書のリストのスナップショットです。このスタティック CRL リストは、認証局 (CA) からデバイスに手動でコピーされます。</p> <p>(注) スタティック CRL は、サポートされている唯一の失効チェック方式です。</p> <p>(注) CRL を削除するには、delete crl コマンドを入力します。</p>
ステップ 7	<p>(任意) show crypto ca certificates</p> <p>例 :</p> <pre>switch(config-trustpoint)# show crypto ca certificates</pre>	設定されている証明書またはチェーンと、関連付けられているトラストポイントを表示します。
ステップ 8	<p>(任意) show crypto ca crl trustpoint</p> <p>例 :</p> <pre>switch(config-trustpoint)# show crypto ca crl winca</pre>	指定したトラストポイントの CRL リストの内容を表示します。
ステップ 9	<p>(任意) show user-account</p> <p>例 :</p> <pre>switch(config-trustpoint)# show user-account</pre>	設定されたユーザアカウントの詳細を表示します。
ステップ 10	<p>(任意) show users</p> <p>例 :</p> <pre>switch(config-trustpoint)# show users</pre>	デバイスにログオンしているユーザが表示されます。
ステップ 11	<p>(任意) copy running-config startup-config</p> <p>例 :</p> <pre>switch(config-trustpoint)# copy running-config startup-config</pre>	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 暗号化アルゴリズムのカスタマイズ

レガシー SSH セキュリティアルゴリズム、メッセージ認証コード (MAC)、キータイプ、および暗号のサポートを設定できます。

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例 : <pre>switch# configure terminal</pre>	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	(任意) ssh kexalgos [all ecdh-sha2-nistp384] 例 : <pre>switch(config)# ssh kexalgos all switch(config)# ssh kexalgos ecdhsha2-nistp384</pre>	<p>接続ごとのキーの生成に使用されるキー交換方式である、サポートされているすべての KexAlgorithms を有効にするには、all キーワードを使用します。</p> <p>サポートされる KexAlgorithms は次のとおりです。</p> <ul style="list-style-type: none"> • curve25519-sha256 • diffie-hellman-group-exchange-sha256 • diffie-hellman-group1-sha1 <p>(注) このアルゴリズムは、Cisco NX-OS リリース 9.3(5)以降ではサポートされていません。SSH クライアントをアップグレードします。</p> <ul style="list-style-type: none"> • diffie-hellman-group14-sha1 • diffie-hellman-group1-sha1 • ecdh-sha2-nistp256 • ecdh-sha2-nistp384 • ecdh-sha2-nistp521 <p>ecdh-sha2-nistp384 KexAlgorithm のみを有効にするには、ecdh-sha2-nistp384 キーワードを使用します。</p>
ステップ 3	(任意) ssh macs all 例 : <pre>switch(config)# ssh macs all switch(config)# ssh macs hmacsha2-256-etm@openssh.com</pre>	<p>トラフィック変更の検出に使用されるメッセージ認証コードである、サポートされているすべての MAC を有効にします。</p> <p>サポートされる MAC は次のとおりです。</p> <ul style="list-style-type: none"> • hmac-sha1

	コマンドまたはアクション	目的
		<ul style="list-style-type: none"> • hmac-sha2-256 • hmac-sha2-512
ステップ 4	<p>(任意) <code>ssh ciphers [all aes256-gcm]</code></p> <p>例 :</p> <pre>switch(config)# ssh ciphers all switch(config)# ssh ciphers aes192-ctr</pre>	<p>サポートされているすべての暗号を有効にして接続を暗号化するには、all キーワードを使用します。</p> <p>サポート対象の暗号方式 :</p> <ul style="list-style-type: none"> • aes128-cbc • aes192-cbc • aes256-cbc • aes128-ctr • aes192-ctr • aes256-ctr • aes256-gcm@openssh.com • aes128-gcm@openssh.com <p>aes256-gcm暗号だけを有効にするには、aes256-gcm キーワードを使用します。</p> <p>(注) aes256-gcm を有効にする前に、ssh cipher-mode weak が無効になっていることを確認します。</p>
ステップ 5	<p>(任意) <code>ssh keytypes all</code></p> <p>例 :</p> <pre>switch(config)# ssh keytypes all switch(config)# ssh keytypes ecdsa-sha2-nistp256</pre>	<p>サーバがクライアントに対して自身を認証するために使用できる公開キーアルゴリズムである、サポートされているすべての <code>PubkeyAcceptedKeyType</code> を有効にします。</p> <p>サポートされるキータイプは次のとおりです。</p> <ul style="list-style-type: none"> • ecdsa-sha2-nistp256 • ecdsa-sha2-nistp384 • ecdsa-sha2-nistp521 • ssh-dss • ssh-rsa

	コマンドまたはアクション	目的
		<p>(注) Cisco NX-OS リリース 10.4(2) 以降では、サポートされている任意のキータイプを設定できます。</p> <p>(注) <code>rsa</code>、<code>dsa</code>、および <code>ecdsa</code> キータイプを有効にするには、対応する SSH ホストキーを生成する必要があります。</p> <p>設定例：</p> <pre>switch(config)# ssh key rsa 2048 switch(config)# ssh key dsa switch(config)# ssh key ecdsa 256</pre>

例

show ssh [ciphers | macs | keytypes | kexalogs | version] コマンドを使用して、サポートされているアルゴリズムを確認できます。

show ssh ciphers

Cipher	Status	FIPS
-----	-----	-----
aes128-ctr	permitted	yes
aes192-ctr	denied	yes
aes256-ctr	permitted	yes
aes128-cbc	denied	yes
aes192-cbc	denied	yes
aes256-cbc	denied	yes
aes256-gcm@openssh.com	permitted	yes
aes128-gcm@openssh.com	permitted	yes
chacha20-poly1305@openssh.com	permitted	no

show ssh macMAC

	Status	FIPS
-----	-----	-----
hmac-sha2-256-etm@openssh.com	permitted	no
hmac-sha2-512-etm@openssh.com	permitted	no
hmac-sha1-etm@openssh.com	permitted	no
hmac-sha2-256	permitted	yes
hmac-sha2-512	permitted	yes
hmac-sha1	permitted	yes
hmac-sha1-96	unsupported	no
hmac-md5	unsupported	no
hmac-md5-96	unsupported	no
umac-64@openssh.com	unsupported	no
umac-128@openssh.com	unsupported	no
hmac-sha1-96-etm@openssh.com	unsupported	no

```

hmac-md5-etm@openssh.com      unsupported    no
umac-64-etm@openssh.com       unsupported    no
umac-128-etm@openssh.com      unsupported    no

```

```

show ssh keytypes Keytype                               Status      FIPS
-----
ecdsa-sha2-nistp256-cert-v01@openssh.com  permitted    no
ecdsa-sha2-nistp384-cert-v01@openssh.com  permitted    no
ecdsa-sha2-nistp521-cert-v01@openssh.com  permitted    no
ssh-rsa-cert-v01@openssh.com              permitted    no
ecdsa-sha2-nistp256                       permitted    yes
ecdsa-sha2-nistp384                       permitted    yes
ecdsa-sha2-nistp521                       permitted    no
rsa-sha2-256                               permitted    no
ssh-rsa                                     permitted    yes
ssh-dss                                     denied       no
ssh-ed25519                                unsupported   no
ssh-ed25519-cert-v01@openssh.com          unsupported   no
ssh-dss-cert-v01@openssh.com              unsupported   no

```

```

show ssh kexalgos
KexAlgorithm                               Status      FIPS
-----
curve25519-sha256                          permitted    no
curve25519-sha256@libssh.org               permitted    no
ecdh-sha2-nistp256                         permitted    yes
ecdh-sha2-nistp384                         permitted    yes
ecdh-sha2-nistp521                         permitted    yes
diffie-hellman-group16-sha512              permitted    yes
diffie-hellman-group14-sha1                permitted    yes
diffie-hellman-group14-sha256              permitted    no

```

```

show ssh version
CiscoSSH 1.9.29, OpenSSH_8.3p1, CiscoSSL 1.1.1t.7.2.500

```

サポートされるアルゴリズム : FIPモードが有効の場合

FIP モードが有効な場合にサポートされるアルゴリズムのリストは次のとおりです。

表 2: サポートされるアルゴリズム : FIPモードが有効の場合

アルゴリズム	サポート対象	サポート対象外
ciphers	<ul style="list-style-type: none"> • aes128-ctr • aes256-ctr • aes256-gcm@openssh.com • aes128-gcm@openssh.com 	<ul style="list-style-type: none"> • aes192-ctr • aes128-cbc • aes192-cbc • aes256-cbc
hmac	<ul style="list-style-type: none"> • hmac-sha2-256 • hmac-sha2-512 • hmac-sha1 	<ul style="list-style-type: none"> • hmac-sha2-256-etm@openssh.com • hmac-sha2-512-etm@openssh.com • hmac-sha1-etm@openssh.com

アルゴリズム	サポート対象	サポート対象外
kexalgo	<ul style="list-style-type: none"> • ecdh-sha2-nistp256 • ecdh-sha2-nistp384 • ecdh-sha2-nistp521 • diffie-hellman-group16-sha512 • diffie-hellman-group14-sha1 • diffie-hellman-group14-sha256 	<ul style="list-style-type: none"> • curve25519-sha256 • curve25519-sha256@libssh.org
keytypes	<ul style="list-style-type: none"> • rsa-sha2-256 • ecdsa-sha2-nistp256 • ecdsa-sha2-nistp384 • ecdsa-sha2-nistp521 	ssh-rsa

デフォルトの SSH サーバポートの変更

Cisco NX-OS Cisco リリース 9.2(1) 以降では、SSHv2 のポート番号をデフォルトのポート番号 22 から変更できます。デフォルトの SSH ポートの変更時に使用される暗号化により、より強力なプライバシーとセッション整合性をサポートする接続が実現します。

手順

	コマンドまたはアクション	目的
ステップ 1	configure terminal 例 : <pre>switch# configure terminal switch(config)#</pre>	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 : <pre>switch(config)# no feature ssh</pre>	SSH を無効にします。
ステップ 3	show sockets local-port-range 例 : <pre>switch(config)# show sockets local port range (15001 - 58000) switch(config)# local port range (58001 - 63535) and nat port range (63536 - 65535) switch# show sockets local-port-range</pre>	使用可能なポート範囲を表示します。

	コマンドまたはアクション	目的
	Kstack local port range (15001 - 22002) Netstack local port range (22003 - 65535)	
ステップ 4	ssh port local-port 例 : <pre>switch(config)# ssh port 58003</pre>	ポートを設定します。 (注) 以前のリリースからリリース 9.3(1) 以降のリリースにアップグレードする場合は、ユーザ定義の SSH ポートを使用する機能が次の範囲内にあることを確認してください。 <ul style="list-style-type: none"> • リリース 9.3(1) およびリリース 9.3(2) の場合 : Kstack ローカルポートの範囲は 15001 ~ 58000、netstack ローカルポートの範囲は 58001 ~ 63535、nat ポートの範囲は 63536 ~ 65535 • リリース 9.3(3) 以降 : Kstack ローカルポートの範囲は 15001 ~ 58000、netstack ローカルポートの範囲は 58001 ~ 60535、nat ポートの範囲は 60536 ~ 65535
ステップ 5	feature ssh 例 : <pre>switch(config)# feature ssh</pre>	SSH を有効にします。
ステップ 6	exit 例 : <pre>switch(config)# exit switch#</pre>	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 7	(任意) show running-config security all 例 : <pre>switch# ssh port 58003</pre>	セキュリティ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 8	(任意) copy running-config startup-config 例 : <pre>switch# copy running-config startup-config</pre>	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。

Procedure

	Command or Action	Purpose
ステップ 1	clear ssh hosts Example: <pre>switch# clear ssh hosts</pre>	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

Cisco NX-OS では、デフォルトで SSH サーバが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。

Procedure

	Command or Action	Purpose
ステップ 1	configure terminal Example: <pre>switch# configure terminal switch(config)#</pre>	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 Example: <pre>switch(config)# no feature ssh</pre>	SSH を無効にします。
ステップ 3	exit Example: <pre>switch(config)# exit switch#</pre>	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。

	Command or Action	Purpose
ステップ 4	(Optional) show ssh server Example: switch# show ssh server	SSH サーバの設定を表示します。
ステップ 5	(Optional) copy running-config startup-config Example: switch# copy running-config startup-config	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。



Note 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。

Procedure

	Command or Action	Purpose
ステップ 1	configure terminal Example: 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 Example: switch(config)# no feature ssh	SSH を無効にします。
ステップ 3	no ssh key[dsa rsa ecdsa] Example: 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 Example: 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	(Optional) show ssh key Example: switch# show ssh key	SSH サーバキーの設定を表示します。

	Command or Action	Purpose
ステップ 6	(Optional) copy running-config startup-config Example: switch# copy running-config startup-config	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。

Related Topics

[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。

Procedure

	Command or Action	Purpose
ステップ 1	show users Example: 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 Example: 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 サーバのイネーブル化

Cisco NX-OS デバイス上で Telnet サーバを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。

Procedure

	Command or Action	Purpose
ステップ 1	configure terminal Example: 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。

	Command or Action	Purpose
ステップ 2	feature telnet Example: 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 Example: 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	(Optional) show telnet server Example: switch# show telnet server	Telnet サーバの設定を表示します。
ステップ 5	(Optional) copy running-config startup-config Example: switch# copy running-config startup-config	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。

Before you begin

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 サーバを有効にします。

リモート デバイス上で Telnet サーバを有効にします。

Procedure

	Command or Action	Purpose
ステップ 1	telnet { <i>ipv4-address</i> <i>host-name</i> } [<i>port-number</i>] [vrf <i>vrf-name</i>] Example: 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。値の範囲は 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 { <i>ipv6-address</i> <i>host-name</i> } [<i>port-number</i>] [vrf <i>vrf-name</i>] Example:	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。値の範囲

	Command or Action	Purpose
	switch# telnet6 2001:0DB8::ABCD:1 vrf management	は 1 ~ 65535 です。デフォルトの VRF はデフォルト VRF です。

Related Topics

[Telnet サーバのイネーブル化](#) (26 ページ)

Telnet セッションのクリア

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

Before you begin

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

Procedure

	Command or Action	Purpose
ステップ 1	show users Example: 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 Example: 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] [md5]	SSH サーバ キーを表示します。 Cisco NX-OS リリース 7.0(3)I4(6) および 7.0(3)I6(1) 以降のリリースでは、このコマンドはデフォルトで SHA256 形式でフィンガープリントを表示します。SHA256 は、以前のデフォルトの MD5 形式よりも安全です。ただし、フィンガープリントを MD5 形式で表示する必要がある場合の下位互換性のために、 md5 オプションが追加されています。
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。 all キーワードを指定すると、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。

コマンド	目的
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 <i>username</i> keypair	指定したユーザの公開キーを表示します。
show user-account	設定されたユーザ アカウントの詳細を表示します。
show users	デバイスにログオンしているユーザが表示されます。
show crypto ca certificates	X.509v3証明書ベースのSSH認証に設定されたCA証明書および関連するトラストポイントを表示します。
show crypto ca crl <i>trustpoint</i>	指定したトラストポイントのCRLリストの内容を表示します。

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。

Procedure

ステップ 1 SSH サーバをディセーブルにします。

Example:

```
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。

Example:

```
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。

Example:

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

ステップ 4 SSH サーバ キーを表示します。

Example:

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

```

*****
rsa Keys generated:Tue Mar 14 13:13:47 2017

ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQGDh4+DZboQJbJt10nJhgKBYL510lhsFM2oZRi9+JqEU
GA44I9ej+E5NIRZ1x8ohIt6Vx9Et5cs07Pw72rjUwR3UPmuAm79k7I/SyLGEp3WUL7sqbLvNF5GqKXph
oqMT075WUdbGWphorA2g0tTObrRfIQBJVQ0SSBh3oEaaALqYUQ==

bitcount:1024
fingerprint:
SHA256:V6KAeLAIKRRUPBZmlYq3rl6JW7Eo7vhLi6CXyxnd/+Y
*****
*****

switch(config)# show ssh key
rsa Keys generated:Sat Sep 29 00:10:39 2013

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。

Example:

```

switch(config)# username User1 sshkey ssh-rsa
AAAAB3NzaC1yc2EAAAABIwAAAIEAy19oF6QaZ19G+3f1XswK30iW4H7YyUyuA50r
v7gsEPjhOBYmsi6PAVKui1nIf/DQhum+1JNqJP/eLowb7ubO+1VKRXYF/G+1JNlQ
W3g9igG30c6k6+XVn+NjnI1B7ihvpVh7dLddMOXwOnXHYshXmSiH3UD/vKyziEh5
4Tplx8=

```

ステップ 6 設定を保存します。

Example:

```

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。

Procedure

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。

Example:

```
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。

Example:

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

*****

rsa Keys generated: Thu Jul  9 11:10:29 2013

ssh-rsa
AAAAB3NzaC1yc2EAAAABIwAAAIEAxWmjJT+oQhIcvnrmBx2BmD0P8boZE1TfJ
Fx9fexWp6rOiztlwODtehnjadWc6A+DE2DvYNvqsrU9TByYDPQkR/+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。

Example:

```
switch(config)# username admin keypair export bootflash:key_rsa rsa
Enter Passphrase:
switch(config)# dir
.
.
.
          951      Jul 09 11:13:59 2013  key_rsa
          221      Jul 09 11:14:00 2013  key_rsa.pub
.
.
```

ステップ 4 これら 2 つのファイルを他の Cisco NX-OS デバイスへコピーした後、**copy scp** または **copy sftp** コマンドを使用して、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。

Example:

```
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 2013

ssh-rsa
```

```

AAAAAB3NzaC1yc2EAAAABIwAAAIEAxWmjJT+oQhIcvnrMbx2BmD0P8boZE1TfJ
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。

Example:

```
$ 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 (Optional) DSA キーについてこの手順を繰り返します。

X.509v3 証明書ベースの SSH 認証の設定例

次の例は、X.509v3 証明書を使用する SSH 認証の設定方法を示しています。



(注) リモート TACACS 認証はサポートされていません。SSH v509v3 証明書ベースの認証のみがサポートされています。

```

configure terminal
username jsmith password 4Ty18Rnt
username jsmith ssh-cert-dn "/O = ABCcompany, OU = ABC1,
emailAddress = jsmith@ABCcompany.com, L = Metropolis, ST = New York, C = US, CN = jsmith"
  rsa
crypto ca trustpoint tp1
crypto ca authenticate tp1
crypto ca crl request tp1 bootflash:crl1.crl

show crypto ca certificates
Trustpoint: tp1
CA certificate 0:
subject= /CN=SecDevCA
issuer= /CN=SecDevCA
serial=01AB02CD03EF04GH05IJ06KL07MN
notBefore=Jun 29 12:36:26 2016 GMT
notAfter=Jun 29 12:46:23 2021 GMT
SHA1 Fingerprint=47:29:E3:00:C1:C1:47:F2:56:8B:AC:B2:1C:64:48:FC:F4:8D:53:AF

```



```

purposes: sslserver sslclient

show crypto ca crt tpt
Trustpoint: tpt CRL: Certificate Revocation List (CRL):
  Version 2 (0x1)
  Signature Algorithm: sha1WithRSAEncryption
  Issuer: /CN=SecDevCA
  Last Update: Aug 8 20:03:15 2016 GMT
  Next Update: Aug 16 08:23:15 2016 GMT
  CRL extensions:
    X509v3 Authority Key Identifier:
      keyid:30:43:AA:80:10:FE:72:00:DE:2F:A2:17:E4:61:61:44:CE:78:FF:2A

show user-account
user:user1
  this user account has no expiry date
  roles:network-operator
  ssh cert DN : /C = US, ST = New York, L = Metropolis, O = cisco , OU = csg, CN
= user1; Algo: x509v3-sign-rsa

show users
NAME      LINE      TIME      IDLE      PID      COMMENT
user1     pts/1     Jul 27 18:43  00:03     18796    (10.10.10.1)  session=ssh

```

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	<i>Cisco NX-OS</i> ライセンス ガイド
VRF コンフィギュレーション	『 <i>Cisco Nexus 9000 Series NX-OS Unicast Routing Configuration Guide</i> 』

RFC

RFC	タイトル
RFC 6187	セキュアシェル認証用のX.509v3証明書

MIB

MIB	MIB のリンク
SSH および Telnet に関連する MIB	サポートされている 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。 ftp://ftp.cisco.com/pub/mibs/supportlists/nexus9000/Nexus9000MIBSupportList.html

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。