管理アクセス

この章では、Telnet、SSH、および HTTPS（ASDM を使用）経由でシステム管理を行うために Cisco ASA にアクセスする方法と、ユーザを認証および許可する方法、ログイン バナーを作成する方法について説明します。

管理リモート アクセスの設定

ここでは、ASDM 用の ASA アクセス、Telnet または SSH、およびログイン バナーなどのその他のパラメータの設定方法について説明します。

SSH アクセスの設定

クライアント IP アドレスを指定して、ASA に SSH を使用して接続できるユーザを定義するには、次の手順を実行します。次のガイドラインを参照してください。

- また、ASA インターフェイスに SSH アクセスの目的でアクセスするために、ホスト IP アドレスを許可するアクセス ルールは必要ありません。このセクションの手順に従って、SSH アクセスを設定する必要があるだけです。
- ASA への通過ルートとなるインターフェイス以外のインターフェイスへの SSH アクセスはサポートされません。たとえば、SSH ホストが外部インターフェイスにある場合、外部インターフェイスへの直接管理接続のみ開始できます。このルールの例外は、VPN 接続を介した場合のみです。VPN トンネルを介した管理アクセスの設定 （12 ページ）を参照してください。
- ASA は、コンテキスト/単一のモードあたり最大 5 つの同時 SSH 接続と、すべてのコンテキストにまたがり分散された最大 100 の接続を許容します。

- （8.4 以降）SSH デフォルト ユーザ名はサポートされなくなりました。**pix** または **asa** ユーザ名とログイン パスワードで SSH を使用して ASA に接続することができなくなりました。SSH を使用するには、**aaa authentication ssh console LOCAL** コマンドを使用して AAA 認証を設定してから、**username** コマンドを入力してローカル ユーザを定義します。ローカル データベースの代わりに AAA サーバを認証に使用する場合、ローカル認証もバックアップの手段として設定しておくことをお勧めします。

始める前に

- マルチ コンテキスト モードでは、コンテキスト実行スペースで次の手順を実行します。システムからコンテキスト コンフィギュレーションに変更するには、**changeto context** *name* を入力します。

手順

ステップ 1 SSH に必要な RSA キー ペアを生成します（物理 ASA の場合のみ）。

crypto key generate rsa modulus *modulus_size*

例：

```
ciscoasa(config)# crypto key generate rsa modulus 2048
```

ASAv の場合、RSA キー ペアは導入後に自動的に作成されます。

係数の値（ビット単位）は 512、768、1024、2048、3072、または 4096 です。指定するキー係数のサイズが大きいほど、RSA キー ペアの生成にかかる時間は長くなります。2048 文字以上の値を推奨します。

ステップ 2 RSA キーを永続的なフラッシュ メモリに保存します。

write memory

例：

```
ciscoasa(config)# write memory
```

ステップ 3 SSH アクセスに使用できるユーザをローカル データベースに作成します。ユーザ アクセスに AAA サーバを使用することもできますが、ローカル ユーザ名の使用を推奨します。

username *name* [**password** *password*] **privilege** *level*

例：

```
ciscoasa(config)# username admin password Far$cape1999 privilege 15
```

デフォルトの特権レベルは 2 です。0 ～ 15 の範囲でレベルを入力します。15 を指定すると、すべての特権を使用できます。ユーザにパスワード認証ではなく公開キー認証（**ssh authentication**）を強制する場合は、パスワードなしでユーザを作成することを推奨します。

username コマンドで公開キー認証およびパスワードの両方を設定した場合、ユーザはいずれの方法でもログインできます（この手順で AAA 認証を明示的に設定した場合）。**注**：ユーザ名とパスワードを作成しなければならないという事態を回避するため、**username** コマンド **nopassword** オプション**nopassword** オプションでは、任意のパスワードを入力できますが、パスワードなしは不可能です。

ステップ 4 （任意）パスワード認証ではなく公開キー認証のみ、またはこれら両方の認証をユーザに許可し、ASA で公開キーを入力します。

username *name* **attributes**

ssh authentication {**pkf** | **publickey** *key*}

例：

```
ciscoasa(config)# username admin attributes
ciscoasa(config-username)# ssh authentication pkf

Enter an SSH public key formatted file.
End with the word "quit" on a line by itself:
---- BEGIN SSH2 PUBLIC KEY ----
Comment: "4096-bit RSA, converted by xxx@xxx from OpenSSH"
AAAAB3NzaC1yc2EAAAADAQABAAACAQDNUvkgza37lB/Q/fljpLAv1BbyAd5PJCJXh/U4LO
hleR/qgIROjpnFaS7Az8/+sjHmq0qXC5TXkzWihvRZbhefyPhPHCi0hIt4oUF2ZbXESA/8
jUT4ehXIUE7FrChffBBtbD4d9FkV8A2gwZCDJBxEM26ocbZCSTx9QC//wt6E/zRcdoqiJG
p4ECEdDaM+56l+yf73NUigO7wYkqcrzjmI1rZRDLVcqtj8Q9qD3MqsV+PkJGSGiqZwnyIl
QbfYxXHU9wLdWxhUbA/xOjJuZ15TQMa7KLs2u+RtrpQgeTGTffIh6O+xKh93gwTgzaZTK4
CQ1kuMrRdNRzza0byLeYPtSlv6Lv6F6dGtwlqrX5a+w/tV/aw9WUg/rapekKloz3tsPTDe
p866AFzU+Z7pVR1389iNuNJHQS7IUA2m0cciIuCM2we/tVqMPYJl+xgKAkuHDkBlMS4i8b
Wzyd+4EUMDGGZVeO+corKTLWFO1wIUieRkrUaCzjComGYZdzrQT2mXBcSKQNWlSCBpCHsk
/r5uTGnKpCNWfL7vd/sRCHyHKsxjsXR15C/5zgHmCTAaGOuIq0Rjo34+61+70PCtYXebxM
Wwm19e3eH2PudZd+rj1dedfr2/IrislEBRJWGLoR/N+xsvwVVM1Qqw1uL4r99CbZF9NghY
NRxCQOY/7K77II==
---- END SSH2 PUBLIC KEY ----
quit
INFO: Import of an SSH public key formatted file SUCCEEDED.
```

ローカル **username** の場合、パスワード認証ではなく公開キー認証のみ、またはこれら両方の認証を有効にできます。SSH-RSA raw キー（証明書なし）を生成可能な任意の SSH キー生成ソフトウェア（ssh keygen など）を使用して、公開キー/秘密キーのペアを生成できます。ASA で公開キーを入力します。その後、SSH クライアントは秘密キー（およびキー ペアを作成するために使用したパスフレーズ）を使用して ASA に接続します。

pkf キーの場合、PKF でフォーマットされたキーを最大 4096 ビット貼り付けるよう求められます。Base64 形式では大きすぎてインラインで貼り付けることができないキーにはこのフォーマットを使用します。たとえば、ssh keygen を使って 4096 ビットのキーを生成してから PKF に変換し、そのキーに対して **pkf** キーワードが求められるようにすることができます。**注**：フェールオーバーで **pkf** オプションを使用することはできますが、PKF キーは、スタンバイ システムに自動的に複製されません。PKF キーを同期するには、**write standby** コマンドを入力する必要があります。

publickey キーの場合、これは Base64 でエンコードされた公開キーのことです。SSH-RSA raw キー（証明書なし）を生成可能な任意の SSH キー生成ソフトウェア（ssh keygen など）を使用して、キーを生成できます。

ステップ 5 （パスワードアクセスの場合）SSH アクセスのためにローカル（または AAA サーバ）認証を有効にします。

aaa authentication ssh console {**LOCAL** | *server_group* [**LOCAL**]}

例：

```
ciscoasa(config)# aaa authentication ssh console LOCAL
```

このコマンドは、**ssh authentication** コマンドでのユーザ名のローカル公開キー認証には影響しません。ASA では、公開キー認証に対し、ローカル データベースを暗黙的に使用します。このコマンドは、ユーザ名とパスワードにのみ影響します。ローカルユーザが公開キー認証またはパスワードを使用できるようにするには、パスワードアクセスを有効にするため、このコマンドで明示的にローカル認証を設定する必要があります。

ステップ 6 ASA がアドレスまたはサブネットごとに接続を受け入れる IP アドレスと、SSH を使用可能なインターフェイスを特定します。

ssh *source_IP_address mask source_interface*

- *source_interface*：名前付きインターフェイスを指定します。ブリッジグループの場合、ブリッジグループ メンバインターフェイスを指定します。VPN 管理アクセスのみ（VPN トンネルを介した管理アクセスの設定 （12 ページ）を参照してください）の場合、名前付き BVI インターフェイスを指定します。

Telnet と異なり、SSH は最も低いセキュリティ レベルのインターフェイスで実行できます。

例：

```
ciscoasa(config)# ssh 192.168.3.0 255.255.255.0 inside
```

ステップ 7 （任意） ASA がセッションを切断するまでに SSH がアイドル状態を維持する時間の長さを設定します。

ssh timeout *minutes*

例：

```
ciscoasa(config)# ssh timeout 30
```

タイムアウトは 1 ～ 60 分に設定します。デフォルトは 5 分です。デフォルトの期間では一般に短すぎるので、実働前のテストとトラブルシューティングがすべて完了するまでは、長めに設定しておいてください。

ステップ 8 （任意） SSH バージョン 1 または 2 へのアクセスを制限します。デフォルトでは、SSH はバージョン 1 と 2 の両方を許可します。

ssh version *version_number*

例：

```
ciscoasa(config)# ssh version 2
```

ステップ 9 （任意） SSH 暗号の暗号化アルゴリズムを設定します。

ssh cipher encryption {**all** | **fips** | **high** | **low** | **medium** | **custom** *colon-delimited_list_of_encryption_ciphers*}

例：

```
ciscoasa(config)# ssh cipher encryption custom 3des-cbc:aes128-cbc:aes192-cbc
```

デフォルトは **medium** です。

- すべての暗号方式（3des-cbc aes128-cbc aes192-cbc aes256-cbc aes128-ctr aes192-ctr aes256-ctr）を使用する場合は、**all** キーワードを使用します。
- カスタム暗号ストリングを設定する場合は、**custom** キーワードを使用し、各暗号ストリングをコロンで区切って入力します。
- FIPS 対応の暗号方式（aes128-cbc aes256-cbc）のみを使用する場合は、**fips** キーワードを使用します。
- 強度が高の暗号方式のみ（aes256-cbc aes256-ctr）を使用する場合は、**high** キーワードを使用します。
- 強度が低、中、高の暗号方式（3des-cbc aes128-cbc aes192-cbc aes256-cbc aes128-ctr aes192-ctr aes256-ctr）を使用する場合は、**low** キーワードを使用します。
- 強度が中および高の暗号方式（3des-cbc aes128-cbc aes192-cbc aes256-cbc aes128-ctr aes192-ctr aes256-ctr）を使用する場合は、**medium** キーワードを使用します（デフォルト）。

ステップ 10 （任意） SSH 暗号の整合性アルゴリズムを設定します。

ssh cipher integrity {**all** | **fips** | **high** | **low** | **medium** | **custom** *colon-delimited_list_of_integrity_ciphers*}

例：

```
ciscoasa(config)# ssh cipher integrity custom hmac-sha1-96:hmac-md5
```

デフォルトは **medium** です。

- すべての暗号方式（hmac-sha1 hmac-sha1-96 hmac-md5 hmac-md5-96）を使用する場合は、**all** キーワードを使用します。
- カスタム暗号ストリングを設定する場合は、**custom** キーワードを使用し、各暗号ストリングをコロンで区切って入力します。
- FIPS 対応の暗号方式（hmac-sha1）のみを使用する場合は、**fips** キーワードを使用します。
- 強度が高の暗号方式のみ（hmac-sha1）を使用する場合は、**high** キーワードを使用します。

- 強度が低、中、高の暗号方式（hmac-sha1 hmac-sha1-96 hmac-md5 hmac-md5-96）を使用する場合は、**low** キーワードを使用します。
- 強度が中および高の暗号方式（hmac-sha1 hmac-sha1-96）を使用する場合は、**medium** キーワードを使用します（デフォルト）。

ステップ 11 （任意） Diffie-Hellman（DH）キー交換モードを設定します。

ssh key-exchange group {**dh-group1-sha1** | **dh-group14-sha1**}

例：

```
ciscoasa(config)# ssh key-exchange group dh-group14-sha1
```

DH キー交換は、いずれかの当事者単独では決定できない共有秘密を提供します。キー交換を署名およびホストキーと組み合わせることで、ホスト認証が実現します。このキー交換方式により、明示的なサーバ認証が可能となります。DH グループ キー交換方式が指定されないと、DH グループ 1 のキー交換方式が使用されます。DH キー交換方法の使用方法の詳細については、RFC 4253 を参照してください。

例

次に、PKF 形式のキーを使用して認証する例を示します。

```
ciscoasa(config)# crypto key generate rsa modulus 4096
ciscoasa(config)# write memory
ciscoasa(config)# username exampleuser1 password examplepassword1 privilege 15
ciscoasa(config)# username exampleuser1 attributes
ciscoasa(config-username)# ssh authentication pkf
Enter an SSH public key formatted file.
End with the word "quit" on a line by itself:
---- BEGIN SSH2 PUBLIC KEY ----
Comment: "4096-bit RSA, converted by xxx@xxx from OpenSSH"
AAAAB3NzaC1yc2EAAAADAQABAAACAQDNUvkgza37lB/Q/fljpLAv1BbyAd5PJCJXh/U4LO
hleR/qgIROjpnFaS7Az8/+sjHmq0qXC5TXkzWihvRZbhefyPhPHCi0hIt4oUF2ZbXESA/8
jUT4ehXIUE7FrChffBBtbD4d9FkV8A2gwZCDJBxEM26ocbZCSTx9QC//wt6E/zRcdoqiJG
p4ECEdDaM+56l+yf73NUigO7wYkqcrzjmI1rZRDLVcqtj8Q9qD3MqsV+PkJGSGiqZwnyIl
QbfYxXHU9wLdWxhUbA/xOjJuZ15TQMa7KLs2u+RtrpQgeTGTffIh6O+xKh93gwTgzaZTK4
CQ1kuMrRdNRzza0byLeYPtSlv6Lv6F6dGtwlqrX5a+w/tV/aw9WUg/rapekKloz3tsPTDe
p866AFzU+Z7pVR1389iNuNJHQS7IUA2m0cciIuCM2we/tVqMPYJl+xgKAkuHDkBlMS4i8b
Wzyd+4EUMDGGZVeO+corKTLWFO1wIUieRkrUaCzjComGYZdzrQT2mXBcSKQNWlSCBpCHsk
/r5uTGnKpCNWfL7vd/sRCHyHKsxjsXR15C/5zgHmCTAaGOuIq0Rjo34+61+70PCtYXebxM
Wwm19e3eH2PudZd+rj1dedfr2/IrislEBRJWGLoR/N+xsvwVVM1Qqw1uL4r99CbZF9NghY
NRxCQOY/7K77II==
---- END SSH2 PUBLIC KEY ----
quit
INFO: Import of an SSH public key formatted file SUCCEEDED.
ciscoasa(config)# ssh 192.168.1.2 255.255.255.255 inside
```

次の例では、Linux または Macintosh システムの SSH の共有キーを生成して、ASA にインポートします。

1. コンピュータで 4096 ビットの ssh-rsa 公開キーおよび秘密キーを生成します。

```
jcrichton-mac:~ john$ ssh-keygen -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/Users/john/.ssh/id_rsa):
/Users/john/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase): pa$$phrase
Enter same passphrase again: pa$$phrase
Your identification has been saved in /Users/john/.ssh/id_rsa.
Your public key has been saved in /Users/john/.ssh/id_rsa.pub.
The key fingerprint is:
c0:0a:a2:3c:99:fc:00:62:f1:ee:fa:f8:ef:70:c1:f9 john@jcrichton-mac
The key's randomart image is:
+--[ RSA 4096]----+
| .               |
|  o  .           |
|+...  o          |
|B.+.....         |
|.B ..+  S        |
|  =   o          |
|   + . E         |
|  o o            |
| ooooo           |
+-----------------+
```

2. PKF 形式にキーを変換します。

```
jcrichton-mac:~ john$ cd .ssh
jcrichton-mac:.ssh john$ ssh-keygen -e -f id_rsa.pub
---- BEGIN SSH2 PUBLIC KEY ----
Comment: "4096-bit RSA, converted by ramona@rboersma-mac from OpenSSH"
AAAAB3NzaC1yc2EAAAADAQABAAACAQDNUvkgza37lB/Q/fljpLAv1BbyAd5PJCJXh/U4LO
hleR/qgIROjpnDaS7Az8/+sjHmq0qXC5TXkzWihvRZbhefyPhPHCi0hIt4oUF2ZbXESA/8
jUT4ehXIUE7FrChffBBtbD4d9FkV8A2gwZCDJBxEM26ocbZCSTx9QC//wt6E/zRcdoqiJG
p4ECEdDaM+56l+yf73NUigO7wYkqcrzjmI1rZRDLVcqtj8Q9qD3MqsV+PkJGSGiqZwnyIl
QbfYxXHU9wLdWxhUbA/xOjJuZ15TQMa7KLs2u+RtrpQgeTGTffIh6O+xKh93gwTgzaZTK4
CQ1kuMrRdNRzza0byLeYPtSlv6Lv6F6dGtwlqrX5a+w/tV/aw9WUg/rapekKloz3tsPTDe
p866AFzU+Z7pVR1389iNuNJHQS7IUA2m0cciIuCM2we/tVqMPYJl+xgKAkuHDkBlMS4i8b
Wzyd+4EUMDGGZVeO+corKTLWFO1wIUieRkrUaCzjComGYZdzrQT2mXBcSKQNWlSCBpCHsk
/r5uTGnKpCNWfL7vd/sRCHyHKsxjsXR15C/5zgHmCTAaGOuIq0Rjo34+61+70PCtYXebxM
Wwm19e3eH2PudZd+rj1dedfr2/IrislEBRJWGLoR/N+xsvwVVM1Qqw1uL4r99CbZF9NghY
NRxCQOY/7K77IQ==
---- END SSH2 PUBLIC KEY ----
jcrichton-mac:.ssh john$
```

3. キーをクリップボードにコピーします。
4. ASA CLI に接続し、公開キーをユーザ名に追加します。

```
ciscoasa(config)# username test attributes
ciscoasa(config-username)# ssh authentication pkf
Enter an SSH public key formatted file.
End with the word "quit" on a line by itself:
---- BEGIN SSH2 PUBLIC KEY ----
Comment: "4096-bit RSA, converted by ramona@rboersma-mac from OpenSSH"
AAAAB3NzaC1yc2EAAAADAQABAAACAQDNUvkgza37lB/Q/fljpLAv1BbyAd5PJCJXh/U4LO
hleR/qgIROjpnDaS7Az8/+sjHmq0qXC5TXkzWihvRZbhefyPhPHCi0hIt4oUF2ZbXESA/8
jUT4ehXIUE7FrChffBBtbD4d9FkV8A2gwZCDJBxEM26ocbZCSTx9QC//wt6E/zRcdoqiJG
```

```
p4ECEdDaM+56l+yf73NUigO7wYkqcrzjmI1rZRDLVcqtj8Q9qD3MqsV+PkJGSGiqZwnyIl
QbfYxXHU9wLdWxhUbA/xOjJuZ15TQMa7KLs2u+RtrpQgeTGTffIh6O+xKh93gwTgzaZTK4
CQ1kuMrRdNRzza0byLeYPtSlv6Lv6F6dGtwlqrX5a+w/tV/aw9WUg/rapekKloz3tsPTDe
p866AFzU+Z7pVR1389iNuNJHQS7IUA2m0cciIuCM2we/tVqMPYJl+xgKAkuHDkBlMS4i8b
Wzyd+4EUMDGGZVeO+corKTLWFO1wIUieRkrUaCzjComGYZdzrQT2mXBcSKQNWlSCBpCHsk
/r5uTGnKpCNWfL7vd/sRCHyHKsxjsXR15C/5zgHmCTAaGOuIq0Rjo34+61+70PCtYXebxM
Wwm19e3eH2PudZd+rj1dedfr2/IrislEBRJWGLoR/N+xsvwVVM1Qqw1uL4r99CbZF9NghY
NRxCQOY/7K77IQ==
---- END SSH2 PUBLIC KEY ----
quit
INFO: Import of an SSH public key formatted file completed successfully.
```

5. ユーザが ASA に SSH できることを確認（テスト）します。

```
jcrichton-mac:.ssh john$ ssh test@10.86.118.5
The authenticity of host '10.86.118.5 (10.86.118.5)' can't be established.
RSA key fingerprint is 39:ca:ed:a8:75:5b:cc:8e:e2:1d:96:2b:93:b5:69:94.
Are you sure you want to continue connecting (yes/no)? yes
```

次のダイアログボックスが、パスフレーズを入力するために表示されます。

一方、端末セッションでは、以下が表示されます。

```
Warning: Permanently added '10.86.118.5' (RSA) to the list of known hosts.
Identity added: /Users/john/.ssh/id_rsa (/Users/john/.ssh/id_rsa)
Type help or '?' for a list of available commands.
asa>
```

Telnet アクセスの設定

Telnet を使用して ASA にアクセス可能なクライアント IP アドレスを指定するには、次の手順を実行します。次のガイドラインを参照してください。

- また、ASA インターフェイスに Telnet アクセスの目的でアクセスするために、ホスト IP アドレスを許可するアクセスルールは必要ありません。このセクションの手順に従って、Telnet アクセスを設定する必要があるだけです。
- ASA への通過ルートとなるインターフェイス以外のインターフェイスへの Telnet アクセスはサポートされません。たとえば、Telnet ホストが外部インターフェイスにある場合、

外部インターフェイスへの直接 Telnet 接続のみ開始できます。このルールの例外は、VPN 接続を介した場合のみです。VPN トンネルを介した管理アクセスの設定 （12 ページ）を参照してください。

- VPN トンネル内で Telnet を使用する場合を除き、最も低いセキュリティ インターフェイスに対して Telnet は使用できません。
- ASA は、コンテキスト/単一のモードあたり最大 5 つの同時 Telnet 接続と、すべてのコンテキストにまたがり分散された最大 100 の接続を許容します。

始める前に

- マルチ コンテキスト モードでは、コンテキスト実行スペースで次の手順を実行します。システムからコンテキスト コンフィギュレーションに変更するには、**changeto context** *name* を入力します。
- Telnet を使用して ASA CLI にアクセスするには、**password** コマンドで設定したログイン パスワードを入力します。Telnet を使用する前に手動でパスワードを設定する必要があります。

手順

ステップ 1 ASA が指定したインターフェイスのアドレスまたはサブネットごとに接続を受け入れる IP アドレスを特定します。

telnet *source_IP_address mask source_interface*

- *source_interface*：名前付きインターフェイスを指定します。ブリッジ グループの場合、ブリッジ グループ メンバインターフェイスを指定します。VPN 管理アクセスのみ（VPN トンネルを介した管理アクセスの設定 （12 ページ）を参照してください）の場合、名前付き BVI インターフェイスを指定します。

インターフェイスが 1 つしかない場合は、インターフェイスのセキュリティ レベルが 100 である限り、そのインターフェイスにアクセスするように Telnet を設定することができます。

例：

```
ciscoasa(config)# telnet 192.168.1.2 255.255.255.255 inside
```

ステップ 2 ASA がセッションを切断するまで Telnet セッションがアイドル状態を維持する時間の長さを設定します。

telnet timeout *minutes*

例：

```
ciscoasa(config)#  telnet timeout 30
```

タイムアウトは 1 ～ 1440 分に設定します。デフォルトは 5 分です。デフォルトの期間では一般に短すぎるので、実働前のテストとトラブルシューティングがすべて完了するまでは、長めに設定しておいてください。

例

次の例は、アドレスが 192.168.1.2 の内部インターフェイスのホストで ASA にアクセスする方法を示しています。

```
ciscoasa(config)# telnet 192.168.1.2 255.255.255.255 inside
```

次の例は、192.168.3.0 のネットワーク上のすべてのユーザが内部インターフェイス上の ASA にアクセスできるようにする方法を示しています。

```
ciscoasa(config)# telnet 192.168.3.0. 255.255.255.255 inside
```

ASDM の HTTPS アクセスの設定

ASDM を使用するには、HTTPS サーバを有効化し、ASA への HTTPS 接続を許可する必要があります。HTTPS アクセスは工場出荷時のデフォルト設定の一部として有効化されています。ASDM への HTTPS アクセスを設定するには、次の手順を実行します。次のガイドラインを参照してください。

- また、ASA インターフェイスに HTTPS アクセスの目的でアクセスするために、ホスト IP アドレスを許可するアクセスルールは必要ありません。このセクションの手順に従って、HTTPSアクセスを設定する必要があるだけです。ただし、HTTP リダイレクトを設定して HTTP 接続を HTTPS に自動的にリダイレクトするには、HTTP を許可するアクセス ルールを有効化する必要があります。そうしないと、インターフェイスがHTTP ポートをリッスンできません。
- ASA への通過ルートとなるインターフェイス以外のインターフェイスへの管理アクセスはサポートされません。たとえば、管理ホストが外部インターフェイスにある場合、外部インターフェイスへの直接管理接続のみ開始できます。このルールの例外は、VPN 接続を介した場合のみです。VPN トンネルを介した管理アクセスの設定 （12 ページ）を参照してください。
- ASA では、コンテキストごとに最大 5 つの同時 ASDM インスタンスを使用でき、全コンテキスト間で最大 32 の ASDM インスタンスの使用が可能です。

  ASDM セッションでは、2 つの HTTPS 接続が使用されます。一方は常に存在するモニタ用で、もう一方は変更を行ったときにだけ存在する設定変更用です。たとえば、ASDM セッションのシステム制限が 32 の場合、HTTPS セッション数は 64 に制限されます。

始める前に

- マルチ コンテキスト モードでは、コンテキスト実行スペースで次の手順を実行します。システムからコンテキスト コンフィギュレーションに変更するには、**changeto context** *name* を入力します。

手順

ステップ 1 ASA が指定したインターフェイスのアドレスまたはサブネットごとに HTTPS 接続を受け入れる IP アドレスを特定します。

http *source_IP_address mask source_interface*

- *source_interface*：名前付きインターフェイスを指定します。ブリッジ グループの場合、ブリッジ グループ メンバ インターフェイスを指定します。VPN 管理アクセスのみ（VPN トンネルを介した管理アクセスの設定 （12 ページ）を参照してください）の場合、名前付き BVI インターフェイスを指定します。

例：

```
ciscoasa(config)# http 192.168.1.2 255.255.255.255 inside
```

ステップ 2 HTTPS サーバをイネーブルにします。

http server enable [*port*]

例：

```
ciscoasa(config)# http server enable 444
```

デフォルトでは、port は 443 です。ポート番号を変更する場合は、必ず ASDM アクセス URL に変更したポート番号を含めてください。たとえば、ポート番号を 444 に変更する場合は、次の URL を入力します。

https://10.1.1.1:444

例

次の例は、HTTPS サーバを有効化し、アドレスが 192.168.1.2 の内部インターフェイス上のホストで ASDM にアクセスする方法を示しています。

```
ciscoasa(config)# http server enable
ciscoasa(config)# http 192.168.1.2 255.255.255.255 inside
```

次の例は、192.168.3.0/24 のネットワーク上のすべてのユーザが内部インターフェイス上の ASDM にアクセスできるようにする方法を示しています。

```
ciscoasa(config)# http 192.168.3.0 255.255.255.0 inside
```

ASDM アクセスまたはクライアントレス SSL VPN のための HTTP リダイレクトの設定

ASDM またはクライアントレス SSL VPN を使用して ASA に接続するには、HTTPS を使用する必要があります。利便性のために、HTTP 管理接続を HTTPS にリダイレクトすることができます。たとえば、HTTP をリダイレクトすることによって、**http://10.1.8.4/admin/** または **https://10.1.8.4/admin/** と入力し、ASDM 起動ページで HTTPS アドレスにアクセスできます。

IPv4 と IPv6 の両方のトラフィックをリダイレクトできます。

始める前に

通常、ホスト IP アドレスを許可するアクセス ルールは必要ありません。ただし、HTTP リダイレクトのためには、HTTP を許可するアクセス ルールを有効化する必要があります。そうしないと、インターフェイスが HTTP ポートをリッスンできません。

手順

Enable HTTP redirect:

http redirect *interface_name* [*port*]

例：

```
ciscoasa(config)# http redirect outside 88
```

port は、インターフェイスが HTTP 接続のリダイレクトに使用するポートを指定します。デフォルトは 80 です。

VPN トンネルを介した管理アクセスの設定

あるインターフェイスで VPN トンネルが終端している場合、別のインターフェイスにアクセスして ASA を管理するには、そのインターフェイスを管理アクセス インターフェイスとして指定する必要があります。たとえば、outside インターフェイスから ASA に入る場合は、この機能を使用して、ASDM、SSH、Telnet、または SNMP 経由で Inside インターフェイスに接続するか、outside インターフェイスから入るときに Inside インターフェイスに ping を実行できます。

ASA への通過ルートとなるインターフェイス以外のインターフェイスへの VPN アクセスはサポートされません。たとえば、VPN アクセスが外部インターフェイスにある場合、外部インターフェイスへの直接接続のみ開始できます。複数のアドレスを覚える必要がないように、ASA の直接アクセス可能インターフェイスの VPN を有効にし、名前解決を使用してください。

管理アクセスは、IPsec クライアント、IPsec サイト間、Easy VPN、AnyConnect SSL VPN クライアントの VPN トンネル タイプ経由で行えます。

始める前に

別個の管理/データ ルーティング テーブルでのルーティングを考慮すると、VPN の端末インターフェイスと管理アクセスインターフェイスは同じ種類である（つまり両方とも管理専用インターフェイスであるか、通常のデータ インターフェイスである）必要があります。

手順

別のインターフェイスから ASA に入るときにアクセスする管理インターフェイスの名前を指定します。

management-access *management_interface*

Easy VPN およびサイト間トンネルでは、名前付き BVI を指定できます（ルーテッド モード）。

例：

```
ciscoasa(config)# management-access inside
```

Firepower 2100 データ インターフェイスでの FXOS の管理アクセスの設定

データ インターフェイスから Firepower 2100 の FXOS を管理する場合、SSH、HTTPS、および SNMP アクセスを設定できます。この機能は、デバイスをリモート管理する場合、および管理 1/1 を隔離されたネットワークに維持する場合に役立ち、隔離されたネットワーク上の FXOS にアクセスするためのネイティブな方法です。この機能を有効にすると、ローカルアクセスに対し管理 1/1 を使用し続けることができます。この機能を使用しながら FXOS の管理 1/1 からのリモート アクセスは許可できないことに注意してください。この機能には、内部パス（デフォルト）を使用した ASA データ インターフェイスへのトラフィックの転送が必要で、FXOS 管理ゲートウェイを 1 つだけ指定できます。

ASA は、FXOS アクセスに非標準ポートを使用します。標準ポートは同じインタ フェースで ASA が使用するため予約されています。ASA が FXOS にトラフィックを転送するときに、非標準の宛先ポートはプロトコルごとに FXOS ポートに変換されます（FXOS の HTTPS ポートは変更しません）。パケット宛先 IP アドレス（ASA インターフェイス IP アドレス）も、FXOS で使用する内部アドレスに変換されます。送信元アドレスは変更されません。トラフィックを返す場合、ASA は自身のデータ ルーティング テーブルを使用して正しい出力インターフェイスを決定します。管理アプリケーションの ASA データ IP アドレスにアクセスする場合、FXOS ユーザ名を使用してログインする必要があります。ASA ユーザ名は ASA 管理アクセスのみに適用されます。

ASA データ インターフェイスで FXOS 管理トラフィック開始を有効にすることもできます。これは、たとえば、SNMP トラップ、NTP と DNS のサーバ アクセスなどに必要です。デフォルトでは、FXOS 管理トラフィック開始は、DNS および NTP のサーバ通信（スマート ソフトウェア ライセンシング通信で必要）用の ASA 外部インターフェイスで有効になっています。

始める前に

- シングル コンテキスト モードのみ。
- ASA 管理専用インターフェイスは除外します。
- ASA データ インターフェイスに VPN トンネルを使用して、FXOS に直接アクセスすることはできません。SSH の回避策として、ASA に VPN 接続し、*ASA* CLI にアクセスし、**connect fxos** コマンドを使用して FXOS CLI にアクセスします。SSH、HTTPS、および SNMPv3 は暗号化できるため、データ インターフェイスへの直接接続は安全です。

手順

ステップ 1 FXOS リモート管理を有効にします。

fxos {**https** | **ssh** | **snmp**} **permit** {*ipv4_address netmask* | *ipv6_address*/*prefix_length*} *interface_name*

例：

```
ciscoasa(config)# fxos https permit 192.168.1.0 255.255.155.0 inside
ciscoasa(config)# fxos https permit 2001:DB8::34/64 inside
ciscoasa(config)# fxos ssh permit 192.168.1.0 255.255.155.0 inside
ciscoasa(config)# fxos ssh permit 2001:DB8::34/64 inside
```

ステップ 2 （任意） サービスのデフォルトのポートを変更します。

fxos {**https** | **ssh** | **snmp**} **port** *port*

次のデフォルトを参照してください。

- HTTPS デフォルト ポート：3443
- SNMP デフォルト ポート：3061
- SSH デフォルト ポート：3022

例：

```
ciscoasa(config)# fxos https port 6666
ciscoasa(config)# fxos ssh port 7777
```

ステップ 3 FXOS が ASA インターフェイスから管理接続を開始できるようにします。

ip-client *interface_name*

デフォルトでは、外部インターフェイスは有効になっています。

例：

```
ciscoasa(config)# ip-client outside
ciscoasa(config)# ip-client services
```

ステップ 4 管理 1/1 上の Firepower Chassis Manager に接続します（デフォルトでは、https://192.168.45.45、ユーザ名：**admin**、パスワード：**Admin123**）。

ステップ 5 [Platform Settings] タブをクリックし、[SSH]、[HTTPS]、または [SNMP] を有効にします。

SSH と HTTPS はデフォルトで有効になっています。

ステップ 6 [Platform Settings] タブで、管理アクセスを許可するように [Access List] を設定します。デフォルトでは、SSH および HTTPS は管理 1/1 192.168.45.0 ネットワークのみを許可します。ASA の [FXOS Remote Management] 設定で指定したアドレスを許可する必要があります。

コンソール タイムアウトの変更

コンソール タイムアウトでは、接続を特権 EXEC モードまたはコンフィギュレーション モードにしておくことができる時間を設定します。タイムアウトに達すると、セッションはユーザ EXEC モードになります。デフォルトでは、セッションはタイムアウトしません。この設定は、コンソール ポートへの接続を保持できる時間には影響しません。接続がタイムアウトすることはありません。

手順

特権セッションが終了するまでのアイドル時間を分単位（0 ～ 60）で指定します。

console timeout *number*

例：

```
ciscoasa(config)# console timeout 0
```

デフォルトのタイムアウトは 0 であり、セッションがタイムアウトしないことを示します。

CLI プロンプトのカスタマイズ

プロンプトに情報を追加する機能により、複数のモジュールが存在する場合にログインしている ASA を一目で確認することができます。この機能は、フェールオーバー時に、両方の ASA に同じホスト名が設定されている場合に便利です。

マルチ コンテキスト モードでは、システム実行スペースまたは管理コンテキストにログインするときに、拡張プロンプトを表示できます。非管理コンテキスト内では、デフォルトのプロンプト（ホスト名およびコンテキスト名）のみが表示されます。

デフォルトでは、プロンプトに ASA のホスト名が表示されます。マルチ コンテキスト モードでは、プロンプトにコンテキスト名も表示されます。CLI プロンプトには、次の項目を表示できます。

cluster-unit	クラスタ ユニット名を表示します。クラスタの各ユニットは一意の名前を持つことができます。
コンテキスト	（マルチ モードのみ）現在のコンテキストの名前を表示します。
domain	ドメイン名を表示します。
hostname	ホスト名を表示します。
priority	フェールオーバー プライオリティを [pri]（プライマリ）または [sec]（セカンダリ）として表示します。

state	ユニットのトラフィック通過状態またはロールを表示します。 フェールオーバーの場合、state キーワードに対して次の値が表示されます。 • [act]：フェールオーバーが有効であり、装置ではトラフィックをアクティブに通過させています。 • [stby]：フェールオーバーはイネーブルです。ユニットはトラフィックを通過させていません。スタンバイ、失敗、または他の非アクティブ状態です。 • [actNoFailover]：フェールオーバーは無効であり、装置ではトラフィックをアクティブに通過させています。 • [stbyNoFailover]：フェールオーバーは無効であり、装置ではトラフィックを通過させていません。これは、スタンバイ ユニットでしきい値を上回るインターフェイス障害が発生したときに生じることがあります。 クラスタリングの場合、state キーワードに対して次の値が表示されます。 • **master** • **slave** たとえば、**prompt hostname cluster-unit state** と設定して「ciscoasa/cl2/slave>」と表示された場合、ホスト名が ciscoasa、ユニット名が cl2、状態名が slave です。

手順

次のコマンドを入力して、CLI プロンプトをカスタマイズします。

prompt {[**hostname**] [**context**] [**domain**] [**slot**] [**state**] [**priority**] [**cluster-unit**]}

例：

```
ciscoasa(config)# prompt hostname context slot state priority
ciscoasa/admin/pri/act(config)#
```

キーワードを入力する順序によって、プロンプト内の要素の順序が決まります。要素はスラッシュ（/）で区切ります。

ログイン バナーの設定

ユーザが ASA に接続するとき、ログインする前、または特権 EXEC モードに入る前に表示されるメッセージを設定できます。

始める前に

- セキュリティの観点から、バナーで不正アクセスを防止することが重要です。「ウェルカム」や「お願いします」などの表現は侵入者を招き入れているような印象を与えるので使用しないでください。以下のバナーでは、不正アクセスに対して正しい表現を設定しています。

```
You have logged in to a secure device.
If you are not authorized to access this device,
log out immediately or risk possible criminal consequences.
```

- バナーが追加された後、次の場合に ASA に対する Telnet または SSH セッションが終了する可能性があります。
  - バナー メッセージを処理するためのシステム メモリが不足している場合。
  - バナー メッセージの表示を試みたときに、TCP 書き込みエラーが発生した場合。
- バナー メッセージのガイドラインについては、RFC 2196 を参照してください。

手順

ユーザが最初に接続したとき（「今日のお知らせ」（motd））、ユーザがログインしたとき（login）、ユーザが特権 EXEC モードにアクセスしたとき（exec）のいずれかに表示するバナーを追加します。

banner {**exec** | **login** | **motd**} *text*

例：

```
ciscoasa(config)# banner motd Welcome to $(hostname).
```

ユーザが ASA に接続すると、まず「今日のお知らせ」バナーが表示され、その後にログインバナーとプロンプトが表示されます。ユーザが ASA に正常にログインすると、exec バナーが表示されます。

複数の行を追加する場合は、各行の前に **banner** コマンドを追加します。

バナー テキストに関する注意事項：

- スペースは使用できますが、CLI を使用してタブを入力することはできません。
- バナーの長さの制限は、RAM およびフラッシュ メモリに関するもの以外はありません。
- ASA のホスト名またはドメイン名は、**$(hostname)** 文字列と **$(domain)** 文字列を組み込むことによって動的に追加できます。
- システム コンフィギュレーションでバナーを設定する場合は、コンテキスト コンフィギュレーションで **$(system)** 文字列を使用することによって、コンテキスト内でそのバナー テキストを使用できます。

例

以下に、「今日のお知らせ」バナーを追加する例を示します。

```
ciscoasa(config)# banner motd Welcome to $(hostname).

ciscoasa(config)# banner motd Contact me at admin@example.com for any issues.
```

管理セッション クォータの設定

ASA で許可する ASDM、SSH、および Telnet の同時最大セッション数を設定できます。この最大値に達すると、それ以降のセッションは許可されず、syslog メッセージが生成されます。システム ロックアウトを回避するために、管理セッション割り当て量のメカニズムではコンソール セッションをブロックできません。

始める前に

マルチ コンテキスト モードでは、 システム実行スペースで次の手順を実行します。 コンテキストをシステム コンフィギュレーションに入力し、**changeto system** コマンドを入力します。

手順

ステップ 1 次のコマンドを入力します。

quota management-session *number*

- *number*：0（無制限）～ 10000 のセッションの集約数を設定します。

例：

例：

```
ciscoasa(config)# quota management-session 1000
```

ステップ 2 使用中の現在のセッションを表示します。

show quota management-session

例：

```
ciscoasa(config)# show quota management-session

quota management-session limit 3
quota management-session warning level 2
quota management-session level 0
quota management-session high water 2
quota management-session errors 0
quota management-session warnings 0
```

システム管理者用 AAA の設定

この項では、システム管理者の認証、管理許可、コマンド許可を設定する方法について説明します。

管理認証の設定

CLI および ASDM アクセスの認証を設定します。

管理認証について

ASA へのログイン方法は、認証を有効にしているかどうかによって異なります。

SSH 認証の概要

認証ありまたは認証なしでの SSH アクセスについては、次の動作を参照してください。

- 認証なし：SSH は認証なしでは使用できません。
- 認証あり：SSH 認証を有効にした場合は、AAA サーバまたはローカルユーザデータベースに定義されているユーザ名とパスワードを入力します。公開キーの認証では、ASA はローカル データベースのみをサポートします。 SSH 公開キー認証を設定した場合、ASA ではローカルデータベースを暗黙的に使用します。ログインにユーザ名とパスワードを使用する場合に必要なのは、SSH認証を明示的に設定することのみです。ユーザEXECモードにアクセスします。

Telnet 認証の概要

認証の有無にかかわらず、Telnet アクセスについては、次の動作を参照してください。

- 認証なし：Telnet の認証を有効にしていない場合は、ユーザ名を入力しません。ログインパスワード（**password** コマンドで設定）を入力します。デフォルトのパスワードはありません。したがって、ASA へ Telnet 接続するには、パスワードを設定する必要があります。ユーザ EXEC モードにアクセスします。

- 認証あり：Telnet認証を有効にした場合は、AAAサーバまたはローカルユーザデータベースに定義されているユーザ名とパスワードを入力します。ユーザEXECモードにアクセスします。

ASDM 認証の概要

認証ありまたは認証なしでのASDMアクセスに関しては、次の動作を参照してください。AAA認証の有無にかかわらず、証明書認証を設定することも可能です。

- 認証なし：デフォルトでは、ブランクのユーザ名と **enable password** コマンドによって設定されたイネーブル パスワード（デフォルトではブランク）を使用して ASDM にログインできます。空白のままにしないように、できるだけ早くイネーブルパスワードを変更することをお勧めします。ホスト名、ドメイン名、およびイネーブル パスワードと Telnet パスワードの設定を参照してください。ログイン画面で（ユーザ名をブランクのままにしないで）ユーザ名とパスワードを入力した場合は、ASDM によってローカル データベースで一致がチェックされることに注意してください。
- 証明書認証（シングル、ルーテッド モードのみ）：ユーザに有効な証明書を要求できます。証明書のユーザ名とパスワードを入力すると、ASA が PKI トラストポイントに対して証明書を検証します。
- AAA 認証：ASDM（HTTPS）認証を有効にした場合は、AAA サーバまたはローカル ユーザ データベースに定義されているユーザ名とパスワードを入力します。これで、ブランクのユーザ名とイネーブル パスワードで ASDM を使用できなくなりました。
- AAA 認証と証明書認証の併用（シングル、ルーテッド モードのみ）：ASDM（HTTPS）認証を有効にした場合は、AAA サーバまたはローカル ユーザ データベースに定義されているユーザ名とパスワードを入力します。証明書認証用のユーザ名とパスワードが異なる場合は、これらも入力するように求められます。ユーザ名を証明書から取得してあらかじめ入力しておくよう選択できます。

シリアル認証の概要

認証ありまたは認証なしでのシリアル コンソール ポートへのアクセスに関しては、次の動作を参照してください。

- 認証なし：シリアルアクセスの認証を有効にしていない場合は、ユーザ名、パスワードを入力しません。ユーザ EXEC モードにアクセスします。
- 認証あり：シリアルアクセスの認証を有効にした場合は、AAAサーバまたはローカルユーザデータベースで定義されているユーザ名とパスワードを入力します。ユーザEXECモードにアクセスします。

enable 認証の概要

ログイン後に特権EXECモードに入るには、**enable**コマンドを入力します。このコマンドの動作は、認証がイネーブルかどうかによって異なります。

- 認証なし：enable 認証を設定していない場合は、**enable** コマンドを入力するときにシステム イネーブル パスワード（**enable password** コマンドで設定）を入力します。デフォルトは空白です。ただし、enable 認証を使用しない場合、**enable** コマンドを入力した後は、特定のユーザとしてログインしていません。これにより、コマンド認可などユーザベースの各機能が影響を受けることがあります。ユーザ名を維持するには、enable 認証を使用してください。
- 認証あり：enable 認証を設定した場合は、ASA はプロンプトにより AAA サーバまたはローカル ユーザ データベースで定義されているユーザ名とパスワードを要求します。この機能は、ユーザが入力できるコマンドを判別するためにユーザ名が重要な役割を果たすコマンド許可を実行する場合に特に役立ちます。

ローカル データベースを使用する enable 認証の場合は、**enable** コマンドの代わりに **login** コマンドを使用できます。**login** コマンドによりユーザ名が維持されますが、認証をオンにするための設定は必要ありません。

⚠

注意 CLI にアクセスできるユーザや特権 EXEC モードを開始できないようにするユーザをローカル データベースに追加する場合は、コマンド認可を設定する必要があります。コマンド認可がない場合、特権レベルが 2 以上（2 がデフォルト）のユーザは、CLI で自分のパスワードを使用して特権 EXEC モード（およびすべてのコマンド）にアクセスできます。あるいは、認証処理でローカル データベースではなく AAA サーバを使用してログイン コマンドを回避するか、またはすべてのローカル ユーザをレベル 1 に設定することにより、システム イネーブル パスワードを使用して特権 EXEC モードにアクセスできるユーザを制御できます。

ホスト オペレーティング システムから ASA へのセッション

一部のプラットフォームでは、ASA の実行を別のアプリケーションとしてサポートしています（例：Catalyst 6500 の ASASM、Firepower 4100/9300 の ASA）。ホスト オペレーティング システムから ASA へのセッションの場合、接続のタイプに応じてシリアルおよび Telnet 認証を設定できます。たとえば、Firepower 2100 の FXOS では、**connect asa** コマンドはシリアル接続を使用します。

マルチ コンテキスト モードでは、システム コンフィギュレーションで AAA コマンドを設定できません。ただし、Telnet またはシリアル認証を管理コンテキストで設定した場合、認証はこれらのセッションにも適用されます。この場合、管理コンテキストの AAA サーバまたはローカル ユーザ データベースが使用されます。

CLIおよびASDM アクセス認証の設定

始める前に

- Telnet、SSH、または HTTP アクセスを設定します。
- 外部認証の場合は、AAA サーバ グループを設定します。ローカル認証の場合は、ローカル データベースにユーザを追加します。

- HTTP 管理認証では、AAA サーバ グループの SDI プロトコルをサポートしていません。
- この機能は、**ssh authentication** コマンドによるローカル ユーザ名に関する SSH 公開キー認証には影響しません。ASA では、公開キー認証に対し、ローカル データベースを暗黙的に使用します。この機能は、ユーザ名とパスワードにのみ影響します。ローカルユーザが公開キー認証またはパスワードを使用できるようにするには、この手順を使用してローカル認証を明示的に設定し、パスワード アクセスを許可する必要があります。

手順

管理アクセス用のユーザを認証します。

aaa authentication {**telnet** | **ssh** | **http** | **serial**} **console** {**LOCAL** | *server_group* [**LOCAL**]}

例：

```
ciscoasa(config)# aaa authentication ssh console radius_1 LOCAL
ciscoasa(config)# aaa authentication http console radius_1 LOCAL
ciscoasa(config)# aaa authentication serial console LOCAL
```

telnet キーワードは Telnet アクセスを制御します。ASASM の場合、このキーワードは **session** コマンドを使用するスイッチからのセッションにも影響します。**ssh** キーワードは SSH アクセスを制御します（パスワードのみ。公開キー認証では暗黙のうちにローカルデータベースが使用されます）。**http** キーワードは ASDM アクセスを制御します。**serial** キーワードはコンソール ポート アクセスを制御します。ASASM の場合、たとえば、このキーワードは **service-module session** コマンドを使用してスイッチからアクセスする仮想コンソールに影響します。Firepower 2100 の場合、このキーワードは **connect asa** コマンドを使用して FXOS からアクセスする仮想コンソールに影響します。

認証に AAA サーバ グループを使用する場合は、AAA サーバが使用できないときにローカル データベースをフォールバック方式として使用するように ASA を設定できます。サーバ グループ名を指定し、その後に **LOCAL**（大文字と小文字の区別あり）を追加します。ローカルデータベースでは AAA サーバと同じユーザ名およびパスワードを使用することを推奨します。これは、ASA のプロンプトでは、どの方式が使用されているかが示されないためです。**LOCAL** だけを入力して、ローカルデータベースを認証の主要方式として（フォールバックなしで）使用することもできます。

enable コマンド認証の設定（特権 EXEC モード）

ユーザが **enable** コマンドを入力する際に、そのユーザを認証できます。

始める前に

enable 認証の概要 （21 ページ）を参照してください。

手順

ユーザを認証するための次のオプションのいずれかを選択します。

- AAA サーバまたはLOCALデータベースを使用してユーザを認証するには、次のコマンドを入力します。

  aaa authentication enable console {**LOCAL** | *server_group* [**LOCAL**]}

  例：

  ```
  ciscoasa(config)# aaa authentication enable console LOCAL
  ```

  ユーザ名とパスワードの入力を求めるプロンプトがユーザに対して表示されます。

  認証に AAA サーバ グループを使用する場合は、AAA サーバが使用できないときにローカルデータベースをフォールバック方式として使用するようにASAを設定できます。サーバ グループ名を指定し、その後に**LOCAL**（大文字と小文字の区別あり）を追加します。ローカル データベースではAAA サーバと同じユーザ名およびパスワードを使用することを推奨します。これは、ASAのプロンプトでは、どの方式が使用されているかが示されないためです。

  LOCALだけを入力して、ローカルデータベースを認証の主要方式として（フォールバックなしで）使用することもできます。

- ローカルデータベースからユーザとしてログインするには、次のコマンドを入力します。

  login

  例：

  ```
  ciscoasa# login
  ```

  ASAにより、ユーザ名とパスワードの入力を求めるプロンプトが表示されます。パスワードを入力すると、ASA により、ユーザはローカル データベースで指定されている特権レベルに置かれます。

  ユーザは独自のユーザ名とパスワードでログインして特権EXECモードにアクセスすることができるので、システムイネーブルパスワードを全員に提供する必要がなくなります。ユーザがログイン時に特権EXECモード（およびすべてのコマンド）にアクセスできるようにするには、ユーザの特権レベルを 2（デフォルト）～ 15 に設定します。ローカル コマンド認可を設定した場合、ユーザは、その特権レベル以下のレベルに割り当てられているコマンドのみを入力できます。

ASDM 証明書認証の設定

AAA 認証の有無にかかわらず証明書認証を必須にできます。ASA は証明書を PKI トラストポイントに照合して検証します。

始める前に

この機能は、シングル ルーテッド モードでのみサポートされます。

手順

ステップ 1 証明書認証をイネーブルにします。

http authentication-certificate *interface_name*[**match** *certificate_map_name*]

例：

```
ciscoasa(config)# crypto ca certificate map map1 10
ciscoasa(config-ca-cert-map)# subject-name eq www.example.com
ciscoasa(config)# http authentication-certificate outside match map1
```

証明書認証はインターフェイスごとに設定できます。その結果、信頼できるインターフェイスまたは内部インターフェイス上の接続については証明書の提示が不要になります。コマンドを複数回使用すれば、複数のインターフェイス上で証明書認証をイネーブルにできます。

証明書が証明書マップと一致することを要件にするには、**match** キーワードとマップ名を指定します。**crypto ca certificate map** コマンドを使用して、マップを設定します。

ステップ 2 （任意） ASDM で証明書からユーザ名を抽出する際に使用する属性を設定します。

http username-from-certificate{*primary-attr* [*secondary-attr*] | **use-entire-name** | **use-script**} [**pre-fill-username**]

例：

```
ciscoasa(config)# http username-from-certificate CN pre-fill-username
```

デフォルトでは、ASDM は CN OU 属性を使用します。

- *primary-attr* 引数は、ユーザ名の抽出に使用する属性を指定します。*secondary-attr* 引数は、オプションで、ユーザ名を抽出するためにプライマリ属性と一緒に使用する追加の属性を指定します。次の属性を使用できます。
  - C：国
  - CN：共通名
  - DNQ：DN 修飾子
  - EA：電子メール アドレス
  - GENQ：世代修飾子
  - GN：名
  - I：イニシャル
  - L：局所性
  - N：名前

- O：組織
- OU：組織単位
- SER：シリアル番号
- SN：姓
- SP：都道府県
- T：役職
- UID：ユーザ ID
- UPN：ユーザ プリンシパル名

- **use-entire-name** キーワードでは DN 名全体を使用します。
- **use-script** キーワードでは ASDM によって生成された Lua スクリプトを使用します。
- **pre-fill-username** キーワードでは、認証を求めるプロンプトにユーザ名が事前入力されています。そのユーザ名が最初に入力したものと異なる場合、最初のユーザ名が事前入力された新しいダイアログボックスが表示されます。そこに、認証用のパスワードを入力できます。

管理許可による CLI および ASDM アクセスの制限

ASA ではユーザの認証時に管理アクセス ユーザとリモート アクセス ユーザを区別できるようになっています。ユーザ ロールを区別することで、リモート アクセス VPN ユーザやネットワーク アクセス ユーザが ASA に管理接続を確立するのを防ぐことができます。

始める前に

RADIUS または LDAP（マッピング済み）ユーザ

ユーザが LDAP 経由で認証されると、ネイティブ LDAP 属性およびその値が Cisco ASA 属性にマッピングされ、特定の許可機能が提供されます。Cisco VSA CVPN3000-Privilege-Level の値を 0 ～ 15 の範囲で設定した後、ldap map-attributes**ldap map-attributes** コマンドを使用して、LDAP 属性を Cisco VAS CVPN3000-Privilege-Level にマッピングします。

RADIUS IETF の **service-type** 属性が、RADIUS 認証および許可要求の結果として access-accept メッセージで送信される場合、この属性は認証されたユーザにどのタイプのサービスを付与するかを指定するために使用されます。

RADIUS Cisco VSA **privilege-level** 属性（ベンダー ID 3076、サブ ID 220）が access-accept メッセージで送信される場合は、ユーザの権限レベルを指定するために使用されます。

TACACS+ ユーザ

「service=shell」で許可が要求され、サーバは PASS または FAIL で応答します。

ローカル ユーザ

指定したユーザ名に対する **service-type** コマンドを設定します。デフォルトでは、service-type は admin で、**aaa authentication console** コマンドで指定されたすべてのサービスに対してフル アクセスが許可されます。

管理許可の属性

管理許可の AAA サーバ タイプおよび有効な値については、次の表を参照してください。ASA ではこれらの値を使用して管理アクセス レベルを決定します。

Management Level	RADIUS/LDAP の（マッピングされた）属性	TACACS+ 属性	ローカル データベースの属性
[Full Access]： **aaa authentication console** コマンド	Service-Type 6（アドミニストレーティブ）、Privilege-Level 1	PASS、特権レベル 1	admin
[Partial Access]： **aaa authentication console** コマンドで設定すると、CLI または ASDM に対するアクセスが許可されます。ただし、**aaa authentication enable console** コマンドを使用して **enable** 認証を設定する場合、CLI ユーザは **enable** コマンドを使用して特権 EXEC モードにアクセスすることはできません。	Service-Type 7（NAS プロンプト）、Privilege-Level 2 以上 Framed (2) および Login (1) サービス タイプは同様に扱われます。	PASS、特権レベル 2 以上	nas-prompt
[No Access]：管理アクセスが拒否されます。ユーザは **aaa authentication console** コマンドで指定されたいずれのサービスも使用できません（**serial** キーワードは除きます。つまり、シリアル アクセスは許可されます）。リモート アクセス（IPsec および SSL）ユーザは、引き続き自身のリモート アクセス セッションを認証および終了できます。他のすべてのサービス タイプ（ボイス、ファクスなど）も同様に処理されます。	Service-Type 5（アウトバウンド）	FAIL	remote-access

その他のガイドライン

- シリアル コンソール アクセスは管理許可に含まれません。
- この機能を使用するには、管理アクセスに AAA 認証も設定する必要があります。CLIおよびASDM アクセス認証の設定 （22 ページ）を参照してください。
- 外部認証を使用する場合は、この機能をイネーブルにする前に、AAA サーバ グループを設定しておく必要があります。

• HTTP 許可は、シングル ルーテッド モードでのみサポートされます。

手順

ステップ 1 Telnet と SSH の管理許可をイネーブルにします。

aaa authorization exec {**authentication-server** | **LOCAL**} [**auto-enable**]

auto-enable キーワードを使用して、十分な認証特権を持つ管理者が、ログインするときに特権 EXEC モードに自動的に入ることができます。

例：

```
ciscoasa(config)# aaa authentication ssh console RADIUS
ciscoasa(config)# aaa authorization exec authentication-server auto-enable
```

ステップ 2 HTTPS の管理許可をイネーブルにします（ASDM）。

aaa authorization http console {**authentication-server** | **LOCAL**}

例：

```
ciscoasa(config)# aaa authentication http console RADIUS
ciscoasa(config)# aaa authorization http console authentication-server
```

ステップ 3

例

次の例は、LDAP 属性マップを定義する方法を示しています。この例では、セキュリティ ポリシーによって、LDAP によって認証されているユーザが、ユーザ レコードのフィールドまたはパラメータの title と company を、IETF-RADIUS service-type と privilege-level にそれぞれマップすることを指定しています。

```
ciscoasa(config)# ldap attribute-map admin-control
ciscoasa(config-ldap-attribute-map)# map-name title IETF-RADIUS-Service-Type
ciscoasa(config-ldap-attribute-map)# map-name company
```

次の例では、LDAP 属性マップを LDAP AAA サーバに適用します。

```
ciscoasa(config)# aaa-server ldap-server (dmz1) host 10.20.30.1
ciscoasa(config-aaa-server-host)# ldap attribute-map admin-control
```

コマンド認可の設定

コマンドへのアクセスを制御する場合、ASA ではコマンド許可を設定でき、ユーザが使用できるコマンドを決定できます。デフォルトでは、ログインするとユーザ EXEC モードにアクセスでき、最低限のコマンドだけが提供されます。**enable** コマンド（または、ローカルデータベースを使用するときは **login** コマンド）を入力すると、特権 EXEC モードおよびコンフィギュレーション コマンドを含む高度なコマンドにアクセスできます。

次の 2 つのコマンド許可方式のいずれかを使用できます。

- ローカル特権レベル
- TACACS+ サーバ特権レベル

コマンド認可について

コマンド認可を有効にし、承認済みのユーザにのみコマンド入力を許容することができます。

サポートされるコマンド認可方式

次の 2 つのコマンド許可方式のいずれかを使用できます。

- ローカル特権レベル：ASA でコマンド特権レベルを設定します。ローカルユーザ、RADIUS ユーザ、または LDAP ユーザ（LDAP 属性を RADIUS 属性にマッピングする場合）を CLI アクセスについて認証する場合、ASA はそのユーザをローカルデータベース、RADIUS、または LDAP サーバで定義されている特権レベルに所属させます。ユーザは、割り当てられた特権レベル以下のコマンドにアクセスできます。すべてのユーザは、初めてログインするときに、ユーザ EXEC モード（レベル 0 または 1 のコマンド）にアクセスします。ユーザは、特権 EXEC モード（レベル 2 以上のコマンド）にアクセスするために再び **enable** コマンドで認証するか、**login** コマンドでログイン（ローカル データベースに限る）できます。

  （注） ローカルデータベース内にユーザが存在しなくても、また CLI 認証や **enable** 認証がない場合でも、ローカル コマンド許可を使用できます。代わりに、**enable** コマンドを入力するときにシステム イネーブル パスワードを入力すると、ASA によってレベル 15 に置かれます。次に、すべてのレベルのイネーブル パスワードを作成します。これにより、**enable** *n*（2 ～ 15）を入力したときに、ASA によってレベル *n* に置かれるようになります。これらのレベルは、ローカル コマンド許可を有効にするまで使用されません。

- TACACS+ サーバ特権レベル：TACACS+ サーバで、ユーザまたはグループが CLI アクセスについて認証した後で使用できるコマンドを設定します。CLI でユーザが入力するすべてのコマンドは、TACACS+ サーバで検証されます。

セキュリティ コンテキストとコマンド許可

AAA 設定はコンテキストごとに個別であり、コンテキスト間で共有されません。

コマンド許可を設定する場合は、各セキュリティ コンテキストを別々に設定する必要があります。この設定により、異なるセキュリティ コンテキストに対して異なるコマンド許可を実行できます。

セキュリティ コンテキストを切り替える場合、管理者は、ログイン時に指定したユーザ名で許可されるコマンドが新しいコンテキスト セッションでは異なる可能性があることや、新しいコンテキストではコマンド許可がまったく設定されていない可能性があることを念頭に置いてください。コマンド許可がセキュリティ コンテキストによって異なる場合があることを管理者が理解していないと、混乱が生じる可能性があります。この動作は、次の仕組みによってさらに複雑になります。

（注） システム実行スペースでは AAA コマンドがサポートされないため、システム実行スペースではコマンド許可を使用できません。

コマンド権限レベル

デフォルトでは、次のコマンドが特権レベル 0 に割り当てられます。その他のすべてのコマンドは特権レベル 15 に割り当てられます。

- **show checksum**
- **show curpriv**
- **イネーブル化**
- **help**
- **show history**
- **login**
- **logout**
- **pager**
- **show pager**
- **clear pager**
- **quit**
- **show version**

コンフィギュレーション モード コマンドを 15 より低いレベルに移動する場合は、configure コマンドも同じレベルに移動してください。このようにしないと、ユーザはコンフィギュレーション モードに入ることができません。

ローカル コマンド許可の設定

ローカル コマンド許可を使用して、コマンドを 16 の特権レベル（0 ～ 15）の 1 つに割り当てることができます。デフォルトでは、各コマンドは特権レベル 0 または 15 に割り当てられます。各ユーザを特定の特権レベルに定義でき、各ユーザは割り当てられた特権レベル以下のコマンドを入力できます。ASA は、ローカルデータベース、RADIUS サーバ、または LDAP サーバ（LDAP 属性を RADIUS 属性にマッピングする場合）に定義されているユーザ特権レベルをサポートしています。

手順

ステップ 1 特権レベルにコマンドを割り当てます。

privilege [**show** | **clear** | **cmd**] **level** *level* [**mode** {**enable** | **cmd**}] **command** コマンド

例：

```
ciscoasa(config)# privilege show level 5 command filter
```

再割り当てする各コマンドに対してこのコマンドを繰り返します。

このコマンドのオプションは、次のとおりです。

- **show**|**clear**|**cmd**：これらのオプション キーワードを使用すると、コマンドの show、clear、または configure 形式に対してだけ特権を設定できます。コマンドの configure 形式は、通常、未修正コマンド（**show** または **clear** プレフィックスなしで）または **no** 形式として、コンフィギュレーションの変更を引き起こす形式です。これらのキーワードのいずれかを使用しない場合は、コマンドのすべての形式が影響を受けます。
- **level** *level*：0 ～ 15 の重大度。
- **mode** {**enable** | **configure**}：ユーザ EXEC モードまたは特権 EXEC モードおよびコンフィギュレーションモードでコマンドを入力することができ、そのコマンドが各モードで異なるアクションを実行する場合は、それらのモードの特権レベルを個別に設定することができます。
  - **enable**：ユーザ EXEC モードと特権 EXEC モードの両方を指定します。
  - **configure**：**configure terminal** コマンドを使用してアクセスされるコンフィギュレーション モードを指定します。
- **command** *command*：設定しているコマンド。設定できるのは、*main* コマンドの特権レベルだけです。たとえば、すべての **aaa** コマンドのレベルを設定できますが、**aaa authentication** コマンドと **aaa authorization** コマンドのレベルを個別に設定できません。

ステップ 2 （任意） コマンド認可のための AAA ユーザを有効にします。このコマンドを入力しない場合、ASA は、ローカルデータベース ユーザの特権レベルだけをサポートし、他のタイプのユーザをすべてデフォルトでレベル 15 に割り当てます。

aaa authorization exec authentication-server [auto-enable]

例：

```
ciscoasa(config)# aaa authorization exec authentication-server
```

さらに、このコマンドは管理認証を有効にします。管理許可による CLI および ASDM アクセスの制限 （26 ページ）を参照してください。

ステップ 3 ローカルのコマンド特権レベルの使用を有効にします。

aaa authorization command LOCAL

例：

```
ciscoasa(config)# aaa authorization command LOCAL
```

コマンド特権レベルを設定する場合は、このコマンドでコマンド許可を設定しない限り、コマンド許可は実行されません。

例

filter コマンドの形式は次のとおりです。

- **filter**（**configure** オプションにより表されます）
- **show running-config filter**
- **clear configure filter**

特権レベルを形式ごとに個別に設定することができます。または、このオプションを省略してすべての形式に同じ特権レベルを設定することもできます。次は、各形式を個別に設定する方法の例です。

```
ciscoasa(config)# privilege show level 5 command filter
ciscoasa(config)# privilege clear level 10 command filter
ciscoasa(config)# privilege cmd level 10 command filter
```

また、次の例では、すべての filter コマンドを同じレベルに設定する例を示します。

```
ciscoasa(config)# privilege level 5 command filter
```

show privilege コマンドは、形式を分けて表示します。

次の例では、**mode** キーワードの使用方法を示します。**enable** コマンドは、ユーザ EXEC モードから入力する必要があります。一方、**enable password** コマンドは、コンフィギュレーション モードでアクセスでき、最も高い特権レベルが必要です。

```
ciscoasa(config)# privilege cmd level 0 mode enable command enable
```

```
ciscoasa(config)# privilege cmd level 15 mode cmd command enable
ciscoasa(config)# privilege show level 15 mode cmd command enable
```

次の例では、**mode** キーワードを使用する追加コマンド（**configure** コマンド）を示します。

```
ciscoasa(config)# privilege show level 5 mode cmd command configure
ciscoasa(config)# privilege clear level 15 mode cmd command configure
ciscoasa(config)# privilege cmd level 15 mode cmd command configure
ciscoasa(config)# privilege cmd level 15 mode enable command configure
```

（注） この最後の行は、**configure terminal** コマンドに関する行です。

TACACS+ サーバでのコマンドの設定

グループまたは個々のユーザの共有プロファイル コンポーネントとしての Cisco Secure Access Control Server（ACS）TACACS+サーバでコマンドを設定できます。サードパーティのTACACS+サーバの場合は、コマンド許可サポートの詳細については、ご使用のサーバのマニュアルを参照してください。

Cisco Secure ACS バージョン 3.1 でコマンドを設定する場合は、次のガイドラインを参照してください。

- ASA は、シェル コマンドとして許可するコマンドを送信し、TACACS+ サーバでシェル コマンドとしてコマンドを設定します。

  （注） Cisco Secure ACS には、「pix-shell」と呼ばれるコマンド タイプが含まれている場合があります。このタイプは ASA コマンド許可に使用しないでください。

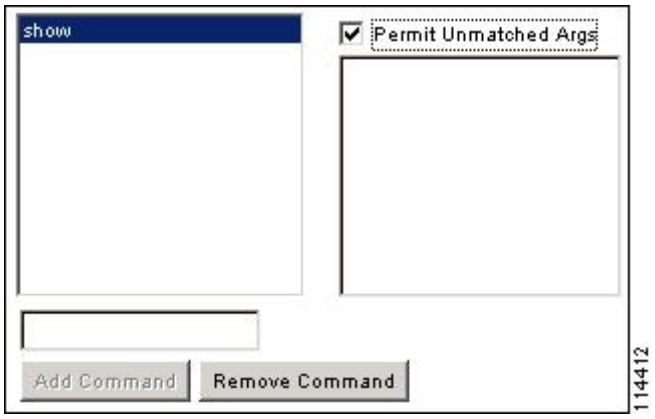
- コマンドの最初のワードは、メイン コマンドと見なされます。その他のワードはすべて引数と見なされます。これは、**permit** または **deny** の後に置く必要があります。

  たとえば、**show running-configuration aaa-server** コマンドを許可するには、コマンド フィールドに **show running-configuration** を追加し、引数フィールドに **permit aaa-server** を入力します。

- [Permit Unmatched Args] チェックボックスをオンにすると、明示的に拒否していないすべてのコマンド引数を許可できます。

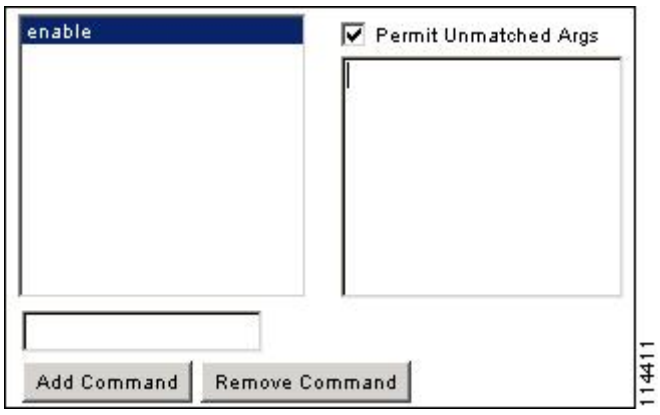
  たとえば、特定の **show** コマンドを設定するだけで、すべての **show** コマンドが許可されます。CLI の使用法を示す疑問符や省略形など、コマンドの変形をすべて予想する必要がなくなるので、この方法を使用することをお勧めします（次の図を参照）。

図 *1*: 関連するすべてのコマンドの許可



- **enable** や **help** など、単一ワードのコマンドについては、そのコマンドに引数がない場合でも、一致しない引数を許可する必要があります（次の図を参照）。

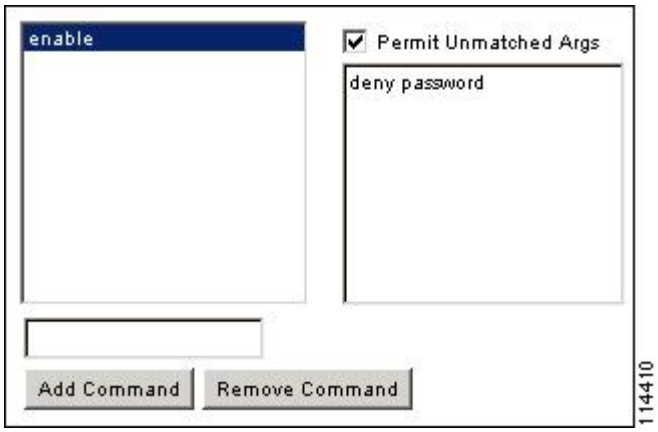
図 *2*: 単一ワードのコマンドの許可



- 引数を拒否するには、その引数の前に **deny** を入力します。

  たとえば、**enable** コマンドを許可し、**enable password** コマンドを許可しない場合には、コマンドフィールドに **enable** を入力し、引数フィールドに **deny password** を入力します。**enable** だけが許可されるように、必ず、[Permit Unmatched Args] チェックボックスをオンにしてください（次の図を参照）。

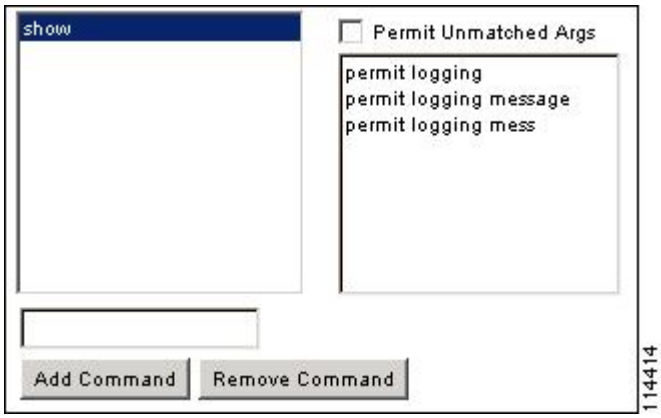
図 *3:* 引数の拒否



- コマンドラインでコマンドを省略形で入力した場合、ASA はプレフィックスとメイン コマンドを完全なテキストに展開しますが、その他の引数は入力したとおりに TACACS+ サーバに送信します。

  たとえば、**sh log** と入力すると、ASA は完全なコマンド **show logging** を TACACS+ サーバに送信します。一方、**sh log mess** と入力すると、ASA は展開されたコマンド **show logging message** ではなく、**show logging mess** を TACACS+ サーバに送信します。省略形を予想して同じ引数の複数のスペルを設定できます（次の図を参照）。

図 *4:* 省略形の指定



- すべてのユーザに対して次の基本コマンドを許可することをお勧めします。
  - **show checksum**
  - **show curpriv**
  - イネーブル化
  - **help**
  - **show history**
  - **login**

- **logout**
- **pager**
- **show pager**
- **clear pager**
- **quit**
- **show version**

TACACS+ コマンド許可の設定

TACACS+ コマンド認可をイネーブルにし、ユーザが CLI でコマンドを入力すると、ASA はそのコマンドとユーザ名を TACACS+ サーバに送信し、コマンドが認可されているかどうかを判別します。

TACACS+ コマンド許可をイネーブルにする前に、TACACS+ サーバで定義されたユーザとして ASA にログインしていること、および ASA の設定を続けるために必要なコマンド許可があることを確認してください。たとえば、すべてのコマンドが認可された管理ユーザとしてログインする必要があります。このようにしないと、意図せずロックアウトされる可能性があります。

意図したとおりに機能することが確認できるまで、設定を保存しないでください。間違いによりロックアウトされた場合、通常は ASA を再始動することによってアクセスを回復できます。

TACACS+ システムが完全に安定して信頼できることを確認します。必要な信頼性レベルについて、通常は、完全冗長 TACACS+ サーバ システムと ASA への完全冗長接続が必要です。たとえば、TACACS+ サーバ プールに、インターフェイス 1 に接続された 1 つのサーバとインターフェイス 2 に接続された別のサーバを含めます。TACACS+ サーバが使用できない場合にフォールバック方式としてローカル コマンド許可を設定することもできます。

TACACS+ サーバを使用したコマンド許可を設定するには、次の手順を実行します。

手順

次のコマンドを入力します。

aaa authorization command *tacacs+_server_group* [**LOCAL**]

例：

```
ciscoasa(config)# aaa authorization command tacacs+_server_group [LOCAL]
```

TACACS+ サーバを使用できない場合は、ローカル データベースをフォールバック方式として使用するように ASA を設定できます。フォールバックを有効にするには、サーバ グループ名の後ろに **LOCAL** を指定します（**LOCAL** は大文字と小文字を区別します）。ローカル データベースでは TACACS+ サーバと同じユーザ名およびパスワードを使用することを推奨します。

これは、ASA のプロンプトでは、どの方式が使用されているかが示されないためです。必ずローカル データベースのユーザとコマンド特権レベルを設定してください。

ローカル データベース ユーザのパスワード ポリシーの設定

ローカル データベースを使用して CLI または ASDM アクセスの認証を設定する場合は、指定期間を過ぎるとユーザにパスワードの変更を要求し、パスワードの最短長と最低変更文字数などのパスワード標準に従うことを要求するパスワード ポリシーを設定できます。

パスワード ポリシーはローカル データベースを使用する管理ユーザに対してのみ適用されます。ローカル データベースを使用するその他のタイプのトラフィック（VPN や AAA によるネットワーク アクセスなど）や、AAA サーバによって認証されたユーザには適用されません。

パスワードポリシーの設定後は、自分または別のユーザのパスワードを変更すると、新しいパスワードに対してパスワードポリシーが適用されます。既存のパスワードについては、現行のポリシーが適用されます。新しいポリシーは、 **username** コマンドおよび **change-password** コマンドを使用したパスワードの変更に適用されます。

始める前に

- ローカルデータベースを使用して CLI または ASDM アクセスの AAA 認証を設定します。
- ローカル データベース内にユーザ名を指定します。

手順

ステップ 1 （オプション）リモート ユーザのパスワードの有効期間を日数で設定します。

password-policy lifetime *days*

例：

```
ciscoasa(config)# password-policy lifetime 180
```

（注） コンソール ポートを使用しているユーザは、パスワードの有効期限が切れてもロックアウトされません。

有効な値は、0 ～ 65536 です。デフォルト値は 0 日です。この場合、パスワードは決して期限切れになりません。

パスワードの有効期限が切れる 7 日前に、警告メッセージが表示されます。パスワードの有効期限が切れると、リモート ユーザのシステム アクセスは拒否されます。有効期限が切れた後アクセスするには、次のいずれかの手順を実行します。

- 他の管理者に **username** コマンドを使用してパスワードを変更してもらいます。
- 物理コンソール ポートにログインして、パスワードを変更します。

ステップ 2 （オプション）新しいパスワードと古いパスワードで違わなければならない最小文字数を設定します。

password-policy minimum-changes *value*

例：

```
ciscoasa(config)# password-policy minimum-changes 2
```

有効な値は、0 ～ 64 文字です。デフォルト値は 0 です

文字マッチングは位置に依存しません。したがって、新しいパスワードで使用される文字が、現在のパスワードのどこにも使用されていない場合に限り、パスワードが変更されたとみなされます。

ステップ 3 （オプション）パスワードの最小長を設定します。

password-policy minimum-length *value*

例：

```
ciscoasa(config)# password-policy minimum-length 8
```

有効な値は、3 ～ 64 文字です。推奨されるパスワードの最小長は 8 文字です。

ステップ 4 （オプション）パスワードに含める大文字の最小個数を設定します。

password-policy minimum-uppercase *value*

例：

```
ciscoasa(config)# password-policy  minimum-uppercase 3
```

有効な値は、0 ～ 64 文字です。デフォルト値は、最小個数がないことを意味する 0 です。

ステップ 5 （オプション）パスワードに含める小文字の最小個数を設定します。

password-policy minimum-lowercase *value*

例：

```
ciscoasa(config)# password-policy minimum-lowercase 6
```

有効な値は、0 ～ 64 文字です。デフォルト値は、最小個数がないことを意味する 0 です。

ステップ 6 （オプション）パスワードに含める数字の最小個数を設定します。

password-policy minimum-numeric *value*

例：

```
ciscoasa(config)# password-policy minimum-numeric 1
```

有効な値は、0 ～ 64 文字です。デフォルト値は、最小個数がないことを意味する 0 です。

ステップ 7 （オプション）パスワードに含める特殊文字の最小個数を設定します。

password-policy minimum-special *value*

例：

```
ciscoasa(config)# password-policy minimum-special 2
```

有効な値は、0 ～ 64 文字です。特殊文字には、!、@、#、$、%、^、&、*、(、および) が含まれます。デフォルト値は、最小個数がないことを意味する 0 です。

ステップ 8 パスワードを再利用を禁止します。

password-policy reuse-interval *value*

例：

```
ciscoasa(config)# password-policy reuse-interval 5
```

以前に使用された 2 ～ 7 個のパスワードと一致するパスワードの再利用を禁止することができます。以前のパスワードは、**password-history** コマンドを使用して、暗号化された形で各ユーザ名の設定に保存されます。このコマンドをユーザが設定することはできません。

ステップ 9 ユーザ名と一致するパスワードを禁止します。

password-policy username-check

ステップ 10 （オプション）ユーザが自分のパスワードの変更に **username** コマンドではなく **change-password** コマンドを使用する必要があるかを設定します。

password-policy authenticate enable

例：

```
ciscoasa(config)# password-policy authenticate enable
```

デフォルト設定はディセーブルです。どちらの方法でも、ユーザはパスワードを変更することができます。

この機能を有効にして、**username** コマンドを使用してパスワードを変更しようとすると、次のエラー メッセージが表示されます。

```
ERROR: Changing your own password is prohibited
```

clear configure username コマンドを使用して自分のアカウントを削除することもできません。消去を試みた場合は、次のエラー メッセージが表示されます。

```
ERROR: You cannot delete all usernames because you are not allowed to delete yourself
```

パスワードの変更

パスワードポリシーでパスワードの有効期間を設定した場合、有効期間を過ぎるとパスワードを新しいパスワードに変更する必要があります。パスワードポリシー認証をイネーブルにした場合は、このパスワード変更のスキームが必須です。パスワードポリシー認証がイネーブルでない場合は、このメソッドを使用することも、直接ユーザアカウントを変更することもできます。

username パスワードを変更するには、次の手順を実行します。

手順

次のコマンドを入力します。

change-password [**old-password** *old_password* [**new-password** *new_password*]]

例：

```
ciscoasa# change-password old-password j0hncr1chton new-password a3rynsun
```

コマンドに新旧のパスワードを入力していない場合は、ASA によって入力が求められます。

ログインの履歴を有効にして表示する

デフォルトでは、ログイン履歴は 90 日間保存されます。この機能を無効にするか、期間を最大 365 日まで変更できます。

始める前に

- ログイン履歴はユニット（装置）ごとに保存されます。フェールオーバーおよびクラスタリング環境では、各ユニットが自身のログイン履歴のみを保持します。
- ログインの履歴データは、リロードされると保持されなくなります。
- 1 つ以上の CLI 管理方式（SSH、Telnet、シリアル コンソール）でローカル AAA 認証をイネーブルにした場合、AAA サーバのユーザ名またはローカル データベースのユーザ名にこの機能が適用されます。ASDM のログインは履歴に保存されません。

手順

ステップ 1 ログインの履歴の期間を次のように設定します。

aaa authentication login-history duration *days*

例：

```
ciscoasa(config)# aaa authentication login-history duration 365
```

days を 1 ～ 365 日に設定できます。デフォルトは 90 です。ログイン履歴を無効にするには、**no aaa authentication login-history** を入力します。

ユーザがログインすると、以下の SSH の例のように、自身のログイン履歴が表示されます。

```
cugel@10.86.194.108's password:
User cugel logged in to ciscoasa at 21:04:10 UTC Dec 14 2016
Last login: 21:01:44 UTC Dec 14 2016 from ciscoasa console
Successful logins over the last 90 days: 6
Authentication failures since the last login: 0
Type help or '?' for a list of available commands.
ciscoasa>
```

ステップ 2 ログイン履歴を次のように表示します。

show aaa login-history [**user** *name*]

例：

```
ciscoasa(config)# show aaa login-history
Login history for user:    turjan
Logins in last   1 days:   1
Last successful login:     16:44:32 UTC Jul 23 2018 from console
Failures since last login: 0
Last failed login:         None
```

管理アクセス アカウンティングの設定

CLI で **show** コマンド以外のコマンドを入力する場合、アカウンティング メッセージを TACACS+ アカウンティング サーバに送信できます。ユーザがログインするとき、ユーザが **enable** コマンドを入力するとき、またはユーザがコマンドを発行するときのアカウンティングを設定できます。

コマンド アカウンティングに使用できるサーバは、TACACS+ だけです。

管理アクセスおよびイネーブル コマンド アカウンティングを設定するには、次の手順を実行します。

手順

ステップ 1 次のコマンドを入力します。

aaa accounting {**serial** | **telnet** | **ssh** | **enable**} **console** *server-tag*

例：

```
ciscoasa(config)# aaa accounting telnet console group_1
```

有効なサーバ グループ プロトコルは RADIUS と TACACS+ です。

ステップ 2 コマンドアカウンティングをイネーブルにします。TACACS+サーバだけがコマンドアカウンティングをサポートします。

aaa accounting command [**privilege** *level*] *server-tag*

例：

```
ciscoasa(config)# aaa accounting command privilege 15 group_1
```

privilege *level* というキーワードと引数のペアは最小特権レベルであり、*server-tag* 引数は ASA がコマンド アカウンティング メッセージを送信する TACACS+ サーバ グループの名前です。

ロックアウトからの回復

状況によっては、コマンド許可やCLI認証をオンにすると、ASA CLIからロックアウトされる場合があります。通常は、ASAを再起動することによってアクセスを回復できます。ただし、すでにコンフィギュレーションを保存した場合は、ロックアウトされたままになる可能性があります。

次の表に、一般的なロックアウト条件とその回復方法を示します。

表 *1: CLI* 認証およびコマンド許可のロックアウト シナリオ

機能	ロックアウト条件	説明	対応策：シングル モード	対応策：マルチ モード
ローカル CLI 認証	ローカル データベースにユーザが設定していない。	ローカル データベース内にユーザが存在しない場合は、ログインできず、ユーザの追加もできません。	ログインし、パスワードと **aaa** コマンドをリセットします。	スイッチからASAへのセッションを接続します。システム実行スペースから、コンテキストに切り替えてユーザを追加することができます。

機能	ロックアウト条件	説明	対応策：シングル モード	対応策：マルチ モード
TACACS+ コマンド許可 TACACS+ CLI 認証 RADIUS CLI 認証	サーバがダウンしているか到達不能で、フォールバック方式を設定していない。	サーバが到達不能である場合は、ログインもコマンドの入力もできません。	1. ログインし、パスワードと AAA コマンドをリセットします。 2. サーバがダウンしたときにロックアウトされないように、ローカル データベースをフォールバック方式として設定します。	1. ASA でネットワーク コンフィギュレーションが正しくないためにサーバが到達不能である場合は、スイッチから ASA へのセッションを接続します。システム実行スペースから、コンテキストに切り替えてネットワークを再設定することができます。 2. サーバがダウンしたときにロックアウトされないように、ローカル データベースをフォールバック方式として設定します。
TACACS+ コマンド許可	十分な特権のないユーザまたは存在しないユーザとしてログインした。	コマンド許可がイネーブルになりますが、ユーザはこれ以上コマンドを入力できなくなります。	TACACS+ サーバのユーザ アカウントを修正します。 TACACS+ サーバへのアクセス権がなく、ASA をすぐに設定する必要がある場合は、メンテナンス パーティションにログインして、パスワードと **aaa** コマンドをリセットします。	スイッチから ASA へのセッションを接続します。システム実行スペースから、コンテキストに切り替えてコンフィギュレーションの変更を完了することができます。また、TACACS+ コンフィギュレーションを修正するまでコマンド許可をディセーブルにすることもできます。
ローカル コマンド許可	十分な特権のないユーザとしてログインしている。	コマンド許可がイネーブルになりますが、ユーザはこれ以上コマンドを入力できなくなります。	ログインし、パスワードと **aaa** コマンドをリセットします。	スイッチから ASA へのセッションを接続します。システム実行スペースから、コンテキストに切り替えてユーザ レベルを変更することができます。

デバイス アクセスのモニタリング

デバイス アクセスのモニタリングについては、次のコマンドを参照してください。

- **show running-config all privilege all**

  このコマンドは、すべてのコマンドの特権レベルを表示します。

  show running-config all privilege all コマンドの場合、ASA は特権レベルに対する各 CLI コマンドの現在の割り当てを表示します。次に、このコマンドの出力例を示します。

```
ciscoasa(config)# show running-config all  privilege all
privilege show level 15 command aaa
privilege clear level 15 command aaa
privilege configure level 15 command aaa
privilege show level 15 command aaa-server
privilege clear level 15 command aaa-server
privilege configure level 15 command aaa-server
privilege show level 15 command access-group
privilege clear level 15 command access-group
privilege configure level 15 command access-group
privilege show level 15 command access-list
privilege clear level 15 command access-list
privilege configure level 15 command access-list
privilege show level 15 command activation-key
privilege configure level 15 command activation-key
...
```

- **show running-config privilege level** *level*

  このコマンドは、特定の特権レベルのコマンドを示します。level 引数は、0 ～ 15 の範囲の整数になります。

  次の例は、特権レベル 10 に対するコマンド割り当てを示しています。

```
ciscoasa(config)# show running-config all privilege level 10
privilege show level 10 command aaa
```

- **show running-config privilege command** コマンド

  このコマンドは、特定のコマンドの特権レベルを表示します。

  次の例は、**access-list** コマンドに対するコマンド割り当てを示しています。

```
ciscoasa(config)# show running-config all privilege command access-list
privilege show level 15 command access-list
privilege clear level 15 command access-list
privilege configure level 15 command access-list
```

- **show curpriv**

  このコマンドは、現在のログイン ユーザを表示します。

  次に、**show curpriv** コマンドの出力例を示します。

```
ciscoasa# show curpriv
Username: admin
Current privilege level: 15
Current Mode/s: P_PRIV
```

次の表で、**show curpriv** コマンドの出力について説明します。

表 *2 : show curpriv* コマンド出力の説明

フィールド	説明
[Username]	[Username]。デフォルト ユーザとしてログインすると、名前は enable_1（ユーザ EXEC）または enable_15（特権 EXEC）になります。
Current privilege level	レベルの範囲は 0 ～ 15 です。ローカル コマンド許可を設定してコマンドを中間特権レベルに割り当てない限り、使用されるレベルはレベル 0 と 15 だけです。
Current Modes	使用可能なアクセス モードは次のとおりです。 • P_UNPR：ユーザ EXEC モード（レベル 0 と 1） • P_PRIV：特権 EXEC モード（レベル 2 ～ 15） • P_CONF：コンフィギュレーション モード

- **show quota management-session**

  このコマンドは、使用中の現在のセッションを表示します。

  次に、**show quota management-session** コマンドの出力例を示します。

```
ciscoasa(config)# show quota management-session

quota management-session limit 3
quota management-session warning level 2
quota management-session level 0
quota management-session high water 2
quota management-session errors 0
quota management-session warnings 0
```

- **show aaa login-history** [**user** *name*]

  このコマンドは、ユーザごとのログイン履歴を表示します。

次に、**show aaa login-history** コマンドの出力例を示します。

```
ciscoasa(config)# show aaa login-history
Login history for user:    turjan
Logins in last   1 days:   1
Last successful login:     16:44:32 UTC Jul 23 2018 from console
Failures since last login: 0
Last failed login:         None
```

管理アクセスの履歴

表 *3:* 管理アクセスの履歴

機能名	プラットフォーム リリース	説明
管理アクセス	7.0(1)	この機能が導入されました。 次のコマンドを導入しました。 **show running-config all privilege all**、**show running-config privilege level**、**show running-config privilege command**、**telnet**、**telnet timeout**、**ssh**、**ssh timeout**、**http**、**http server enable**、**asdm image disk**、**banner**、**console timeout**、**icmp**、**ipv6 icmp**、**management access**、**aaa authentication console**、**aaa authentication enable console**、**aaa authentication telnet | ssh console**、**service-type**、**login**、**privilege**、**aaa authentication exec authentication-server**、**aaa authentication command LOCAL**、**aaa accounting serial | telnet | ssh | enable console**、**show curpriv**、**aaa accounting command privilege**。

機能名	プラットフォーム リリース	説明
SSH セキュリティが向上し、SSH デフォルトユーザ名はサポートされなくなりました。	8.4(2)	8.4(2) 以降、pix または asa ユーザ名とログイン パスワードで SSH を使用して ASA に接続することができなくなりました。SSH を使用するには、**aaa authentication ssh console LOCAL** コマンド（CLI）または [Configuration] > [Device Management] > [Users/AAA] > [AAA Access] > [Authentication (ASDM)] を使用して AAA 認証を設定してから、ローカルユーザを定義する必要があります。定義するには、**username** コマンド（CLI）を入力するか、[Configuration] > [Device Management] > [Users/AAA] > [User Accounts (ASDM)] を選択します。ローカルデータベースの代わりに AAA サーバを認証に使用する場合、ローカル認証もバックアップの手段として設定しておくことをお勧めします。
ローカルデータベースを使用する場合の管理者パスワードポリシーのサポート	8.4(4.1)、9.1(2)	ローカルデータベースを使用して CLI または ASDM アクセスの認証を設定する場合は、指定期間を過ぎるとユーザにパスワードの変更を要求し、パスワードの最短長と最低変更文字数などのパスワード標準に従うことを要求するパスワード ポリシーを設定できます。 次のコマンドが導入されました。**change-password**、**password-policy lifetime**、**password-policy minimum changes**、**password-policy minimum-length**、**password-policy minimum-lowercase**、**password-policy minimum-uppercase**、**password-policy minimum-numeric**、**password-policy minimum-special**、**password-policy authenticate enable**、**clear configure password-policy**、**show running-config password-policy**.

機能名	プラットフォーム リリース	説明
SSH 公開キー認証のサポート	8.4(4.1)、9.1(2)	ASA への SSH 接続の公開キー認証は、ユーザ単位で有効にできます。公開キー ファイル（PKF）でフォーマットされたキーまたは Base64 キーを指定できます。PKF キーは、4096 ビットまで使用できます。ASA がサポートする Base64 形式（最大 2048 ビット）では大きすぎるキーについては、PKF 形式を使用します。 次のコマンドが導入されました。**ssh authenticaion**。 *PKF* キー形式のサポートは *9.1(2)* 以降のみです。
SSH キー交換の Diffie-Hellman グループ 14 のサポート	8.4(4.1)、9.1(2)	SSH キー交換に Diffie-Hellman グループ 14 が追加されました。これまでは、グループ 1 だけがサポートされていました。 次のコマンドが導入されました。**ssh key-exchange**。
管理セッションの最大数のサポート	8.4(4.1)、9.1(2)	同時 ASDM、SSH、Telnet セッションの最大数を設定できます。 次のコマンドが導入されました。**quota management-session**、**show running-config quota management-session**、**show quota management-session**。
マルチ コンテキスト モードの ASASM において、スイッチからの Telnet 認証および仮想コンソール認証をサポートしました。	8.5(1)	マルチ コンテキスト モードのスイッチから ASASM への接続はシステム実行スペースに接続しますが、これらの接続を制御するために管理コンテキストでの認証を設定できます。
SSH の AES-CTR 暗号化	9.1(2)	ASA での SSH サーバの実装が、AES-CTR モードの暗号化をサポートするようになりました。

機能名	プラットフォーム リリース	説明
SSH キー再生成間隔の改善	9.1(2)	SSH 接続は、接続時間 60 分間またはデータ トラフィック 1 GB ごとに再生成されます。 次のコマンドが導入されました。**show ssh sessions detail**。
改善されたワンタイムパスワード認証	9.2(1)	十分な認可特権を持つ管理者は、認証クレデンシャルを一度入力すると特権 EXEC モードに移行できます。**auto-enable** オプションが **aaa authorization exec** コマンドに追加されました。 次のコマンドが変更されました。**aaa authorization exec**。
ASDM 管理認証	9.4(1)	HTTP アクセスと Telnet および SSH アクセス別に管理認証を設定できるようになりました。 次のコマンドが導入されました。 **aaa authorization http console**
証明書コンフィギュレーションの ASDM ユーザ名	9.4(1)	ASDM の証明書認証（**http authentication-certificate**）を有効にすると、ASDM が証明書からユーザ名を抽出する方法を設定できます。また、ログインプロンプトでユーザ名を事前に入力して表示できます。 次のコマンドが導入されました。**http username-from-certificate**
HTTP リダイレクトの IPv6 サポート	9.1(7)/9.6(1)	ASDM アクセスまたはクライアントレス SSL VPN 用の HTTPS に HTTP リダイレクトを有効にすると、IPv6 アドレスへ送信されるトラフィックもリダイレクトできるようになりました。 次のコマンドに機能が追加されました。**http redirect**

機能名	プラットフォーム リリース	説明
設定可能なSSH暗号機能と整合性アルゴリズム	9.1(7)/9.4(3)/9.5(3)/9.6(1)	ユーザはSSH暗号化を管理するときに暗号化モードを選択し、さまざまなキー交換アルゴリズムに対してHMACと暗号化を設定できます。アプリケーションに応じて、暗号の強度を強くしたり弱くする必要がある場合があります。セキュアなコピーのパフォーマンスは暗号化アルゴリズムに一部依存します。デフォルトで、ASA は 3des-cbc aes128-cbc aes192-cbc aes256-cbc aes128-ctr aes192-ctr aes256-ctr の順にアルゴリズムをネゴシエートします。提示された最初のアルゴリズム（3des-cbc）が選択された場合、aes128-cbc などの一層効率的なアルゴリズムが選択された場合よりも大幅にパフォーマンスが低下します。たとえば、提示された暗号方式に変更するには、**ssh cipher encryption custom aes128-cbc** を使用します。 次のコマンドが導入されました。**ssh cipher encryption**、**ssh cipher integrity**。
ASDM に対する ASA SSL サーバ モード マッチング	9.6(2)	証明書マップと照合するために、証明書で認証を行うASDMユーザに対して証明書を要求できるようになりました。 次のコマンドを変更しました。**http authentication-certificate match**

機能名	プラットフォーム リリース	説明
SSH 公開キー認証の改善	9.6(2)	以前のリリースでは、ローカルユーザデータベース（**(aaa authentication ssh console LOCAL)**）を使用して AAA SSH 認証を有効にしなくても、SSH 公開キー認証（**(ssh authentication)**）を有効にすることができました。この設定は修正されたため、AAA SSH 認証を明示的に有効にする必要があります。ユーザが秘密キーの代わりにパスワードを使用できないよう、パスワード未定義のユーザ名を作成できるようになりました。 次のコマンドが変更されました。**ssh authentication**、**username**
ログイン履歴	9.8(1)	デフォルトでは、ログイン履歴は 90 日間保存されます。この機能を無効にするか、期間を最大 365 日まで変更できます。1 つ以上の管理メソッド（SSH、ASDM、Telnet など）でローカル AAA 認証を有効にしている場合、この機能はローカル データベースのユーザ名にのみ適用されます。 次のコマンドが導入されました。**aaa authentication login-history**、**show aaa login-history**
パスワードの再利用とユーザ名と一致するパスワードの使用を禁止するパスワード ポリシーの適用	9.8(1)	最大 7 世代にわたるパスワードの再利用と、ユーザ名と一致するパスワードの使用を禁止できるようになりました。 次のコマンドが導入されました。**password-history**、**password-policy reuse-interval**、**password-policy username-check**

機能名	プラットフォーム リリース	説明
SSH公開キー認証を使用するユーザの認証とパスワードを使用するユーザの認証を区別します。	9.6(3)/9.8(1)	9.6(2) より前のリリースでは、ローカル ユーザ データベース（**ssh authentication**）を使用して AAA SSH 認証を明示的に有効にしなくても、SSH 公開キー認証（**aaa authentication ssh console LOCAL**）を有効にすることができました。9.6(2) では、ASA で AAA SSH 認証を明示的に有効にする必要がありました。このリリースでは、AAA SSH 認証を明示的に有効にする必要はありません。ユーザに対して **ssh authentication** コマンドを設定すると、このタイプの認証を使用するユーザのローカル認証がデフォルトで有効になります。さらに、明示的に AAA SSH 認証を設定すると、この設定はパスワード付きのユーザ名にのみ適用されます。また、任意のAAAサーバ タイプ（**aaa authentication ssh console radius_1** など）を使用できます。たとえば、一部のユーザはローカルデータベースを使用して公開キー認証を使用し、他のユーザはRADIUS でパスワードを使用できます。 変更されたコマンドはありません。
RSA キー ペアは 3072 ビット キーをサポートしています	9.9(2)	モジュラス サイズを 3072 に設定できるようになりました。 新規または変更されたコマンド：**crypto key generate rsa modulus**
ブリッジ型仮想インターフェイス（BVI）の VPN 管理アクセス	9.9(2)	VPN の **management-access** がその BVI で有効になっている場合、**telnet**、**http**、**ssh** などの管理サービスを BVI で有効にできるようになりました。非VPN管理アクセスの場合は、ブリッジ グループ メンバ インターフェイスでこれらのサービスの設定を続行する必要があります。 新規または変更されたコマンド：**https**、**telnet**、**ssh**、**management-access**