



## 基本設定

この章では、ASA上でコンフィギュレーションを機能させるために通常必要な基本設定を行う方法について説明します。

- [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 \(1 ページ\)](#)
- [日時の設定 \(4 ページ\)](#)
- [マスターパスフレーズの設定 \(8 ページ\)](#)
- [DNS サーバの設定 \(12 ページ\)](#)
- [ハードウェア バイパス \(Cisco ISA 3000\) の設定 \(14 ページ\)](#)
- [ASP \(高速セキュリティ パス\) のパフォーマンスと動作の調整 \(16 ページ\)](#)
- [DNS キャッシュのモニタリング \(19 ページ\)](#)
- [基本設定の履歴 \(19 ページ\)](#)

## 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

ホスト名、ドメイン名、イネーブルパスワード、Telnet パスワードを設定するには、次の手順を実行します。

### 始める前に

ホスト名、ドメイン名、イネーブルパスワード、Telnet パスワードを設定する前に、次の要件を確認します。

- 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とシステム実行スペースの両方のホスト名とドメイン名を設定できます。
- イネーブルパスワードと Telnet パスワードは、各コンテキストで設定します。システムでは使用できません。マルチ コンテキスト モードのスイッチから ASASM へのセッションを実行する場合、ASASM は管理コンテキストで設定したログインパスワードを使用します。

- システム コンフィギュレーションからコンテキスト コンフィギュレーションに切り替えるには、**changeto context name** コマンドを入力します。

## 手順

**ステップ 1** ASA またはコンテキストのホスト名を指定します。デフォルトのホスト名は「asa」です。

**hostname name**

例 :

```
ciscoasa(config)# hostname myhostnameexample12345
```

名前には、63 文字以下の文字を使用できます。ホスト名はアルファベットまたは数字で開始および終了する必要があります。使用できるのはアルファベット、数字、ハイフンのみです。

ASA のホスト名を設定すると、そのホスト名がコマンドラインのプロンプトに表示されます。このホスト名によって、複数のデバイスとのセッションを確立する場合に、コマンドを入力する場所が常に把握できます。

マルチ コンテキスト モードでは、システム実行スペースに設定したホスト名がすべてのコンテキストのコマンドラインプロンプトに表示されます。コンテキスト内で任意に設定したホスト名はコマンドラインには表示されませんが、**banner** コマンド**\$(hostname)** トークンによって使用できます。

**ステップ 2** ASA のドメイン名を指定します。デフォルト ドメイン名は **default.domain.invalid** です。

**domain-name name**

例 :

```
ciscoasa(config)# domain-name example.com
```

ASA は、修飾子を持たない名前のサフィックスとして、ドメイン名を追加します。たとえば、ドメイン名を「example.com」に設定し、syslog サーバとして非修飾名「jupiter」を指定した場合は、ASA によって名前が修飾されて「jupiter.example.com」となります。

**ステップ 3** イネーブルパスワードを変更します。デフォルトではイネーブルパスワードは空白ですが。

**enable password password**

例 :

```
ciscoasa(config)# enable password Pa$$w0rd
```

enable 認証を設定しない場合、イネーブルパスワードによって特権 EXEC モードが開始されます。HTTP 認証を設定しない場合、イネーブルパスワードによって空のユーザ名で ASDM にログインできます。

`password` 引数は、大文字と小文字が区別される 3 ～ 32 文字のパスワードです。スペースと疑問符を除く任意の ASCII 印刷可能文字（文字コード 32 ～ 126）を組み合わせることができます。

このコマンドによって最高の特権レベル（15）のパスワードが変更されます。ローカルコマンド許可を設定すると、次の構文を使用して 0 ～ 15 の各特権レベルにイネーブルパスワードを設定できます。

**enable password** *password level number*

**encrypted** キーワードは、（MD5 ベースのハッシュまたは PBKDF2（Password-Based Key Derivation Function 2）ハッシュを使用して）パスワードが暗号化されていることを示します。**enable password** コマンドのパスワードを定義すると、ASA はセキュリティを維持するために、そのパスワードを設定に保存するときに暗号化します。**show running-config** コマンドを入力すると、**enable password** コマンドでは実際のパスワードは示されません。暗号化されたパスワードとそれに続けて **encrypted** キーワードが示されます。たとえば、パスワードに「test」と入力すると、**show running-config** コマンドの出力には次のように表示されます。

```
username user1 password DLaUiAX3l78qgoB5c7iVNw== encrypted
```

実際に CLI で **encrypted** キーワードを入力するのは、同じパスワードを使用して、ある設定ファイルを他の ASA で使用するためにカット アンド ペーストする場合だけです。

パスワードを指定せずに **enable password** コマンドを入力すると、パスワードはデフォルトの空白に設定されます。

**ステップ 4** Telnet アクセスのためのログインパスワードを設定します。デフォルトのパスワードはありません。

Telnet 認証を設定しない場合、ログインパスワードは Telnet アクセスに使用されます。**session** コマンドを使用してスイッチから ASASM にアクセスする場合にも、このパスワードを使用します。

**{passwd | password}** *password* [**encrypted**]

例：

```
ciscoasa(config)# password cisco12345
```

**passwd** または **password** と入力できます。*password* は、大文字と小文字が区別されるパスワードです。英数字と特殊記号を 16 文字まで使用できます。パスワードには、疑問符とスペースを除いて、任意の文字を使用できます。

パスワードは暗号化された形式でコンフィギュレーションに保存されるため、パスワードの入力後に元のパスワードを表示することはできません。何らかの理由で別の ASA にパスワードをコピーする必要があるが、元のパスワードがわからない場合、暗号化されたパスワードと、**encrypted** キーワードを指定して **passwd** コマンドを入力できます。通常、このキーワードは、**show running-config passwd** コマンドを入力するときだけにだけ表示されます。

## 日時の設定



(注) ASASM または Firepower 2100、4100、または 9300 の日時を設定しないでください。ASA は シャーシから日時の設定を受信します。

## タイムゾーンと夏時間の日付の設定

タイムゾーンおよび夏時間の日付範囲を設定するには、次の手順を実行します。

### 手順

**ステップ 1** タイムゾーンを設定します。デフォルトでは、タイムゾーンは UTC です。

• **clock timezone zone [-]hours [minutes]**

- *zone* : タイムゾーンを文字列で指定します (太平洋標準時の PST など)。
- *[-]hours* : UTC からのオフセットの時間数を設定します。たとえば、PST は -8 時間です。
- *minutes* : UTC からのオフセットの分数を設定します。

例 :

```
ciscoasa(config)# clock timezone PST -8
```

**ステップ 2** 次のいずれかのコマンドを入力して、夏時間の日付範囲をデフォルトから変更します。デフォルトの定期的な日付範囲は、3 月の第 2 日曜日の午前 2 時 ~ 11 月の第 1 日曜日の午前 2 時です。

- 夏時間の開始日と終了日を、特定の年の特定の日付として指定します。このコマンドを使用する場合は、日付を毎年再設定する必要があります。

**clock summer-time zone date {day month | month day} year hh:mm {day month | month day} year hh:mm [offset]**

- *zone* : タイムゾーンを文字列で指定します (太平洋夏時間の PDT など)。
- *day* : 1 ~ 31 の日付を設定します。標準の日付形式に応じて、月日を **April 1** または **1 April** のように入力できます。
- *month* : 月を文字列で設定します。標準の日付形式に応じて、月日を **April 1** または **1 April** のように入力できます。
- *year* : 年を 4 桁で設定します (2004 など)。年の範囲は 1993 ~ 2035 です。

- *hh:mm* : 時間と分を 24 時間形式で設定します。
- *offset* : 夏時間用に時間を変更する分数を設定します。デフォルト値は 60 分です。

例 :

```
ciscoasa(config)# clock summer-time PDT 1 April 2010 2:00 60
```

- 夏時間の開始日と終了日を、年の特定の日付ではなく、月の日時形式で指定します。このコマンドを使用すると、毎年変更する必要がない、繰り返される日付範囲を設定できます。

**clock summer-time zone recurring** [*week weekday month hh:mm week weekday month hh:mm*] [*offset*]

- *zone* : タイムゾーンを文字列で指定します (太平洋夏時間の PDT など)。
- *week* : 月の特定の週を 1 から 4 までの整数で指定するか、*first* または *last* という単語で指定します。たとえば、日付が 5 週目に当たる場合は、*last* を指定します。
- *weekday* : Monday、Tuesday、Wednesday などのように曜日を指定します。
- *month* : 月を文字列で設定します。
- *hh:mm* : 時間と分を 24 時間形式で設定します。
- *offset* : 夏時間用に時間を変更する分数を設定します。デフォルト値は 60 分です。

例 :

```
ciscoasa(config)# clock summer-time PDT recurring first Monday April 2:00 60
```

## NTP サーバを使用した日付と時刻の設定

NTP を利用して階層的なサーバシステムを実現し、ネットワーク システム間の時刻を正確に同期します。このような精度は、CRL の検証など正確なタイム スタンプを含む場合など、時刻が重要な操作で必要になります。複数の NTP サーバを設定できます。ASA は、データ信頼度の尺度となる一番下のストラタムのサーバを選択します。

手動で設定した時刻はすべて、NTP サーバから取得された時刻によって上書きされます。

### 始める前に

マルチ コンテキスト モードでは、時刻はシステム コンフィギュレーションに対してだけ設定できます。

## 手順

ステップ1 (任意) NTP サーバによる MD5 認証を有効にします。

- a) 認証をイネーブルにします。

**ntp authenticate**

例 :

```
ciscoasa(config)# ntp authenticate
```

NTP 認証を有効にする場合は、さらに **ntp trusted-key** コマンドでキー ID を指定し、そのキーを **ntp server key** コマンドでサーバに関連付ける必要があります。 **ntp authentication-key** コマンドを使用して ID の実際のキーを設定します。複数のサーバがある場合は、サーバごとに個別の ID を設定します。

- b) 認証キー ID が信頼できるキーであると指定します。この信頼できるキーは、NTP サーバでの認証に必要です。

**ntp trusted-key key\_id**

例 :

```
ciscoasa(config)# ntp trusted-key 1
ciscoasa(config)# ntp trusted-key 2
ciscoasa(config)# ntp trusted-key 3
ciscoasa(config)# ntp trusted-key 4
```

*key\_id* 引数は、1 ~ 4294967295 の値です。複数のサーバで使用できるように複数の信頼できるキーを入力できます。

- c) NTP サーバの認証を行うためのキーを設定します。

**ntp authentication-key key\_id md5 key**

例 :

```
ciscoasa(config)# ntp authentication-key 1 md5 aNiceKey1
ciscoasa(config)# ntp authentication-key 2 md5 aNiceKey2
ciscoasa(config)# ntp authentication-key 3 md5 aNiceKey3
ciscoasa(config)# ntp authentication-key 4 md5 aNiceKey4
```

- *key\_id* : **ntp trusted-key** コマンドを使用して設定した ID を設定します。

- **md5 key** : MD5 キーを最大 32 文字の文字列で設定します。

ステップ2 NTP サーバを指定します。

**ntp server ip v4\_address [key key\_id] [source interface\_name] [prefer]**

例 :

```
ciscoasa(config)# ntp server 10.1.1.1 key 1 prefer
```

```
ciscoasa(config)# ntp server 10.2.1.1 key 2
```

NTP 認証 (**ntp authenticate**) をイネーブルにした場合は、**ntp trusted-key** コマンドを使って設定した ID を使用して **key keykey\_id** 引数を指定する必要があります。

**source interface\_name** キーワード引数ペアは、NTP パケットの発信インターフェイスを識別します (ルーティングテーブル内のデフォルトのインターフェイスを使用しない場合)。マルチコンテキストモードではシステムにインターフェイスが含まれないため、管理コンテキストに定義されているインターフェイス名を指定します。

**prefer** キーワードは、精度が類似する複数のサーバがある場合に、この NTP サーバを優先サーバに設定します。NTP では、どのサーバの精度が最も高いかを判断するためのアルゴリズムを使用し、そのサーバに同期します。サーバの精度に差がない場合は、**prefer** キーワードで使用するサーバを指定します。ただし、優先サーバよりも精度が大幅に高いサーバがある場合、ASA は精度の高いそのサーバを使用します。たとえば、ASA は優先サーバであるストラタム 3 のサーバよりもストラタム 2 のサーバを優先的に使用します。

複数のサーバを指定できます。その中から ASA は最も精度の高いサーバを使用します。

## 手動での日時の設定

日付と時刻を手動で設定するには、次の手順を実行します。

### 始める前に

マルチ コンテキスト モードでは、時刻はシステム コンフィギュレーションに対してだけ設定できます。

### 手順

日付と時刻を手動で設定します。

```
clock set hh:mm:ss {month day | day month} year
```

例 :

```
ciscoasa# clock set 20:54:00 april 1 2004
```

*hh:mm:ss* 引数には、時、分、秒を 24 時間形式で設定します。たとえば、午後 8:54 の場合は、20:54:00 と入力します。

*day* 値は、月の日付として 1 ~ 31 を設定します。標準の日付形式に応じて、月日を *april 1* または *1 april* のように入力できます。

*month* 値は、月を設定します。標準の日付形式に応じて、月日を *april 1* または *1 april* のように入力できます。

year 値は、4桁で年を設定します（2004 など）。年の範囲は 1993 ～ 2035 です。

デフォルトの時間帯は UTC です。**clock timezone** コマンドを使用して、**clock set** コマンドの入力後に時間帯を変更した場合、時間は自動的に新しい時間帯に調整されます。

このコマンドはハードウェアチップ内の時間を設定しますが、コンフィギュレーションファイル内の時間は保存しません。この時間はリブート後も保持されます。他の **clock** コマンドとは異なり、このコマンドは特権 EXEC コマンドです。クロックをリセットするには、**clock set** コマンドを使用して新しい時刻を設定する必要があります。

## マスターパスフレーズの設定

マスターパスフレーズを利用すると、プレーンテキストのパスワードが安全に、暗号化形式で保存され、1つのキーを使用してすべてのパスワードを一様に暗号化またはマスキングできるようになります。このようにしても、機能は一切変更されません。マスターパスフレーズを使用する機能としては、次のものがあります。

- OSPF
- EIGRP
- VPN ロード バランシング
- VPN (リモート アクセスおよびサイトツーサイト)
- フェールオーバー
- AAA サーバ
- Logging
- 共有ライセンス

## マスターパスフレーズの追加または変更

マスターパスフレーズを追加または変更するには、次の手順を実行します。

### 始める前に

- この手順を実行できるのは、コンソール、SSH、HTTPS 経由の ASDM などによるセキュアセッションにおいてのみです。
- フェールオーバーがイネーブルであっても、フェールオーバー共有キーが設定されていない場合に、マスターパスフレーズを変更すると、エラーメッセージが表示されます。このメッセージには、マスターパスフレーズの変更がプレーンテキストとして送信されないよう、フェールオーバー共有キーを入力する必要があることが示されます。



- アクティブ/スタンバイ フェールオーバーでパスワードの暗号化を有効化または変更すると、**write standby** が実行されます。これは、アクティブな構成をスタンバイ ユニットに複製します。この複製が行われない場合、スタンバイユニットの暗号化されたパスワードは、同じパスフレーズを使用している場合でも異なるものになります。構成を複製することで、構成が同じであることが保証されます。アクティブ/アクティブ フェールオーバーの場合は、手動で **write standby** を入力する必要があります。**write standby** は、アクティブ/アクティブ モードでトラフィックの中断を引き起こす場合があります。これは、新しい構成が同期される前に、セカンダリ ユニットで構成が消去されるためです。**failover active group 1** および **failover active group 2** コマンドを使用してプライマリ ASA ですべてのコンテキストをアクティブにし、**write standby** を入力してから、**no failover active group 2** コマンドを使用してセカンダリ ユニットにグループ 2 コンテキストを復元する必要があります。

## 手順

- ステップ 1** 暗号キーの生成に使用されるパスフレーズを設定します。パスフレーズの長さは、8 ～ 128 文字にする必要があります。パスフレーズには、バックスペースと二重引用符を除くすべての文字を使用できます。コマンドに新しいパスフレーズを入力しないと、入力を求めるプロンプトが表示されます。パスフレーズを変更するには、古いパスフレーズを入力する必要があります。

**key config-key password-encryption** [*new\_passphrase* [*old\_passphrase*]]

例：

```
ciscoasa(config)# key config-key password-encryption
Old key: bumblebee
New key: haverford
Confirm key: haverford
```

(注) インタラクティブプロンプトを使用してパスワードを入力し、パスワードがコマンド履歴バッファに記録されないようにします。

暗号化されたパスワードがプレーンテキストパスワードに変換されるため、**no key config-key password-encrypt** コマンドは注意して使用してください。パスワードの暗号化がサポートされていないソフトウェアバージョンにダウングレードするときは、このコマンドの **no** 形式を使用できます。

- ステップ 2** パスワード暗号化をイネーブルにします。

**password encryption aes**

例：

```
ciscoasa(config)# password encryption aes
```

パスワードの暗号化がイネーブルになり、マスターパスワードが使用可能になると、ただちにすべてのユーザパスワードが暗号化されます。実行コンフィギュレーションには、パスワードは暗号化された形式で表示されます。

パスワードの暗号化をイネーブルにしたときに、パスフレーズが設定されていない場合、パスフレーズが将来的に使用可能になるものとしてコマンドは正常に実行されます。

後から **no password encryption aes** コマンドを使用してパスワードの暗号化をディセーブルにすると、暗号化された既存のパスワードは変更されず、マスターパスフレーズが存在する限り、暗号化されたパスワードはアプリケーションによって必要に応じて復号化されます。

**ステップ 3** マスターパスフレーズのランタイム値と結果のコンフィギュレーションを保存します。

#### write memory

例：

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

このコマンドを入力しなければ、スタートアップコンフィギュレーションのパスワードは引き続き可読状態となります（過去に暗号化された状態で保存されていない場合）。また、マルチコンテキストモードでは、マスターパスフレーズはシステムコンテキストコンフィギュレーション内で変更されます。その結果、すべてのコンテキスト内のパスワードが影響を受けます。すべてのユーザコンテキストではなく、システムコンテキストモードで **write memory** コマンドを入力しないと、ユーザコンテキストで暗号化されたパスワードは失効する可能性があります。また、すべての設定を保存するには、システムコンテキストで **write memory all** コマンドを使用します。

---

例

次の例は、これまでにキーが何も存在していないことを示します。

```
ciscoasa(config)# key config-key password-encryption 12345678
```

次の例は、キーがすでに存在することを示します。

```
ciscoasa(config)# key config-key password-encryption 23456789  
Old key: 12345678
```

次の例では、パラメータを指定しないでコマンドを入力して、キーの入力を求めるプロンプトが表示されるようにします。キーがすでに存在するため、入力を求めるプロンプトが表示されます。

```
ciscoasa(config)# key config-key password-encryption  
Old key: 12345678  
New key: 23456789  
Confirm key: 23456789
```

次の例では、既存のキーがないため、入力を求めるプロンプトが表示されません。

```
ciscoasa(config)# key config-key password-encryption
New key: 12345678
Confirm key: 12345678
```

## マスターパスワードの無効化

マスターパスワードをディセーブルにすると、暗号化されたパスワードがプレーンテキストパスワードに戻ります。暗号化されたパスワードをサポートしていない以前のソフトウェアバージョンにダウングレードする場合は、パスワードを削除しておく便利です。

### 始める前に

- ディセーブルにする現在のマスターパスワードがわかっていなければなりません。パスワードが不明の場合は、[マスターパスワードの削除 \(12 ページ\)](#) を参照してください。
- この手順が機能するのは、HTTPS を介した Telnet、SSH、または ASDM によるセキュアセッションだけです。

マスターパスワードをディセーブルにするには、次の手順を実行します。

### 手順

**ステップ 1** マスターパスワードを削除します。コマンドにパスワードを入力しないと、入力を求めるプロンプトが表示されます。

**no key config-key password-encryption** [*old\_passphrase*]

例：

```
ciscoasa(config)# no key config-key password-encryption

Warning! You have chosen to revert the encrypted passwords to plain text.
This operation will expose passwords in the configuration and therefore
exercise caution while viewing, storing, and copying configuration.

Old key: bumblebee
```

**ステップ 2** マスターパスワードのランタイム値と結果のコンフィギュレーションを保存します。

**write memory**

例：

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

パスワードを含む不揮発性メモリは消去され、0xFF パターンで上書きされます。

マルチモードでは、システム コンテキスト コンフィギュレーション内のマスター パスフレーズが変更されます。その結果、すべてのコンテキスト内のパスワードが影響を受けます。すべてのユーザ コンテキストではなく、システム コンテキスト モードで `write memory` コマンドを入力すると、ユーザ コンテキストで暗号化されたパスワードは失効する可能性があります。また、すべての設定を保存するには、システム コンテキストで `write memory all` コマンドを使用します。

---

## マスター パスフレーズの削除

マスター パスフレーズは回復できません。マスター パスフレーズがわからなくなった場合や不明な場合は、削除できます。

マスター パスフレーズを削除するには、次の手順を実行します。

### 手順

- ステップ 1** マスターキーと、暗号化されたパスワードが含まれているコンフィギュレーションを削除します。

**write erase**

例：

```
ciscoasa(config)# write erase
```

- ステップ 2** マスター キーや暗号化パスワードのないスタートアップ コンフィギュレーションを使用して ASA をリロードします。

**reload**

例：

```
ciscoasa(config)# reload
```

---

## DNS サーバの設定

DNS サーバを設定して、ASA がホスト名を IP アドレスに解決できるようにする必要があります。また、アクセスルールに完全修飾ドメイン名 (FQDN) ネットワーク オブジェクトを使用するように、DNS サーバを設定する必要があります。

一部の ASA 機能では、ドメイン名で外部サーバにアクセスするために DNS サーバを使用する必要があります。たとえば、ボットネットトラフィックフィルタ機能では、ダイナミックデー

データベースサーバにアクセスして、スタティックデータベースのエントリを解決するためにDNSサーバが必要です。他の機能（ping コマンドやtracertコマンドなど）では、ping やtracertを実行する名前を入力できるため、ASAはDNSサーバと通信することで名前を解決できます。名前は、多くのSSL VPN コマンドおよびcertificate コマンドでもサポートされます。



- (注) ASA では、機能に応じて DNS サーバの使用が限定的にサポートされます。たとえば、ほとんどのコマンドでは、IP アドレスを入力する必要があります。名前を使用できるのは、名前と IP アドレスを関連付けるように **name** コマンドを手動で設定し、**names** コマンドを使用して名前の使用を有効にした場合だけです。

### 始める前に

DNS ドメインルックアップをイネーブルにするすべてのインターフェイスに対して適切なルーティングおよびアクセスルールを設定し、DNS サーバに到達できるようにしてください。

### 手順

- ステップ 1** サポートされているコマンドに対してネームルックアップを実行するために、ASA が DNS サーバに DNS 要求を送信できるようにします。

**dns domain-lookup interface\_name**

例：

```
ciscoasa(config)# dns domain-lookup inside
```

- ステップ 2** ASA が発信要求に使用する DNS サーバグループを指定します。

**dns server-group DefaultDNS**

例：

```
ciscoasa(config)# dns server-group DefaultDNS
```

PN トンネルグループ用に他の DNS サーバグループを設定できます。詳細については、コマンドリファレンスの **tunnel-group** コマンドを参照してください。

- ステップ 3** 1つまたは複数のDNSサーバを指定します。同じコマンドで6つのIPアドレスすべてをスペースで区切って入力するか、各コマンドを別々に入力できます。ASAでは、応答を受信するまで各DNSサーバを順に試みます。

**name-server ip\_address [ip\_address2] [...] [ip\_address6]**

例：

```
ciscoasa(config-dns-server-group)# name-server 10.1.1.5 192.168.1.67 209.165.201.6 dmz
```

**ステップ4** ホスト名に追加するドメイン名を設定します (完全修飾されていない場合)。

**domain-name** *name*

例 :

```
ciscoasa (config-dns-server-group) # domain-name example.com
```

**ステップ5** (任意) DNS サーバグループの追加プロパティを設定します。

デフォルト設定がネットワークに適さない場合は、次のコマンドを使用してグループの特性を変更します。

- **timeout seconds** : 次の DNS サーバを試行する前に待機する秒数 (1 ~ 30)。デフォルト値は 2 秒です。ASA がサーバのリストを再試行するたびに、このタイムアウトは倍増します。
- **retries number** : ASA が応答を受信しないときに、DNS サーバのリストを再試行する回数 (0 ~ 10)。
- **expire-entry-timer minutes number** : DNS エントリの期限が切れた (TTL が経過した) 後、そのエントリが DNS ルックアップテーブルから削除されるまでの分数。エントリを削除するとテーブルの再コンパイルが必要になります。このため、頻繁に削除するとデバイスの処理負荷が大きくなる可能性があります。DNS エントリによっては TTL が極端に短い (3 秒程度) 場合があるため、この設定を使用して TTL を実質的に延長できます。デフォルトは 1 分です (つまり、TTL が経過してから 1 分後にエントリが削除されます)。指定できる範囲は 1 ~ 65535 分です。このオプションは、FQDN ネットワーク オブジェクトの解決時にのみ使用されます。
- **poll-timer minutes number** : FQDN ネットワーク/ホスト オブジェクトを IP アドレスに解決するために使用されるポーリングサイクルの時間 (分単位)。FQDN オブジェクトはファイアウォールポリシーで使用される場合にのみ解決されます。タイマーによって解決間隔の最大時間が決まります。IP アドレス解決に対して更新するタイミングの決定には DNS エントリの存続可能時間 (TTL) 値も使用されるため、個々の FQDN がポーリングサイクルよりも頻繁に解決される場合があります。デフォルトは 240 (4 時間) です。指定できる範囲は 1 ~ 65535 分です。

---

## ハードウェアバイパス (Cisco ISA 3000) の設定

ハードウェアバイパスを有効化して、停電時にもインターフェイスペア間のトラフィックのフローを継続することができます。サポートされているインターフェイスペアは、銅線 GigabitEthernet 1/1 と 1/2 および GigabitEthernet 1/3 と 1/4 です。ハードウェアバイパスがアクティブな場合はファイアウォール機能が設定されていません。したがって、トラフィックの通過を許可しているリスクをご自身が理解していることを確認してください。次のハードウェアバイパスのガイドラインを参照してください。

- この機能は、Cisco ISA 3000 アプライアンスのみで使用できます。
- 光ファイバーサネット モデルがある場合は、銅線イーサネット ペア (GigabitEthernet 1/1 および 1/2) のみがハードウェアバイパスをサポートします。
- ISA 3000 への電源が切断され、ハードウェアバイパス モードに移行すると、通信できるのはサポートされているインターフェイスペアだけになります。つまり、デフォルトの設定を使用している場合、inside1 と inside2 間および outside1 と outside2 間は通信できなくなります。これらのインターフェイス間の既存の接続がすべて失われます。
- シスコでは、TCPシーケンスのランダム化を無効にすることを推奨しています (下記の手順を参照)。ランダム化が有効化されている場合 (デフォルト)、ハードウェアバイパスを有効化するときに TCPセッションを再確立する必要があります。デフォルトでは、ISA 3000 を通過する TCP 接続の最初のシーケンス番号 (ISN) が乱数に書き換えられます。ハードウェアバイパスが有効化されると、ISA 3000 はデータパスに存在しなくなり、シーケンス番号を変換しません。受信するクライアントは予期しないシーケンス番号を受信し、接続をドロップします。TCPシーケンスのランダム化が無効になっていても、スイッチオーバーの際に一時的にダウンしたリンクのために、一部の TCP 接続は再確立される必要があります。
- ハードウェアのバイパス インターフェイスでの Cisco TrustSec の接続は、ハードウェアのバイパスが有効化されているときにはドロップされます。ISA 3000 の電源がオンになり、ハードウェアのバイパスが非アクティブ化されている場合、接続は再ネゴシエートされません。
- ハードウェアバイパスを非アクティブ化し、トラフィックが ISA 3000 のデータパスを経由することを再開した場合、スイッチオーバー時に一時的にダウンしたリンクがあるために、既存の TCPセッションの一部を再確立する必要があります。
- ハードウェアバイパスをアクティブにすると、イーサネット PHY が切断され、ASA はインターフェイスのステータスを判断できなくなります。インターフェイスはダウン状態であるかのように表示されます。

### 始める前に

- ハードウェアバイパス インターフェイスはスイッチのアクセス ポートに接続する必要があります。トランク ポートには接続しないでください。

### 手順

**ステップ 1** 停電時にハードウェアバイパスが有効化されるように設定します。

**hardware-bypass GigabitEthernet {1/1-1/2 | 1/3-1/4} [sticky]**

例 :

```
ciscoasa(config)# hardware-bypass GigabitEthernet 1/1-1/2
ciscoasa(config)# hardware-bypass GigabitEthernet 1/3-1/4
```

**sticky** キーワードによって、電源が回復してアプライアンスが起動した後に、アプライアンスがハードウェアバイパスモードに保たれます。この場合、準備が整った時点でハードウェアバイパスを手動でオフにする必要があります。このオプションを使用すると、トラフィックへの短時間の割り込みがいつ発生するかを制御できます。

**ステップ2** 手動でハードウェアバイパスを有効化または非アクティブ化します。

**[no] hardware-bypass manual GigabitEthernet {1/1-1/2 |1/3-1/4}**

例：

```
ciscoasa# hardware-bypass manual GigabitEthernet 1/1-1/2
ciscoasa# no hardware-bypass manual GigabitEthernet 1/1-1/2
```

**ステップ3** （任意）ハードウェアバイパスを設定して、ASA FirePOWER モジュールが起動するまでアクティブに維持します。

**hardware-bypass boot-delay module-up sfr**

ブート遅延が動作するには、**sticky** オプションを使用せずにハードウェアバイパスを有効化する必要があります。**hardware-bypass boot-delay** を使用しないと、ASA FirePOWER モジュールが起動を完了する前にハードウェアバイパスが非アクティブになる可能性があります。たとえば、モジュールをフェールクローズに設定していた場合、このような状況では、トラフィックがドロップされる可能性があります。

**ステップ4** TCPシーケンスのランダム化のディセーブルこの例では、デフォルト設定に設定を追加することによって、すべてのトラフィックのランダム化を無効化する方法を示します。

**policy-map global\_policy**

**class sfrclass**

**set connection random-sequence-number disable**

後でオンに戻す場合は、「disable」を **enable** に置き換えます。

---

## ASP（高速セキュリティパス）のパフォーマンスと動作の調整

ASPはポリシーおよび設定を利用可能にする実装レイヤです。Cisco Technical Assistance Centerとのトラブルシューティング時以外は直接影響することはありません。ただし、パフォーマンスと信頼性に関連するいくつかの動作を調節することができます。

### ルールエンジンのトランザクションコミットモデルの選択

デフォルトでは、ルールベースのポリシー（アクセスルールなど）を変更した場合、変更はただちに有効になります。ただし、この即時性によりパフォーマンスにわずかな負担がかかります。



す。パフォーマンスコストは、1秒あたりの接続数が多い環境で大量のルールリストがある場合に顕著です。たとえば、ASAが1秒あたり18,000個の接続を処理しながら、25,000個のルールがあるポリシーを変更する場合などです。

ルールエンジンはさらに迅速なルールルックアップを実現するためにルールをコンパイルするため、パフォーマンスに影響します。デフォルトでは、システムは接続試行の評価時にコンパイルされていないルールも検索して、新しいルールが適用されるようにします。ルールがコンパイルされていないため、検索に時間がかかります。

この動作を変更して、ルールエンジンがトランザクションモデルを使用してルールの変更を導入し、新しいルールがコンパイルされて使用可能な状態になるまで古いルールを引き続き使用するようにできます。トランザクションモデルを使用することで、ルールのコンパイル中にパフォーマンスが落ちることはありません。次の表は、その動作の違いを明確にします。

モデル	コンパイル前	コンパイル中	コンパイル後
デフォルト	古いルールに一致します。	新しいルールに一致します (接続数/秒のレートは減少します)。	新しいルールに一致します。
トランザクション	古いルールに一致します。	古いルールに一致します (接続数/秒のレートは影響を受けません)。	新しいルールに一致します。

トランザクションモデルのその他のメリットには、インターフェイス上のACLを交換するとき、古いACLを削除して新しいポリシーを適用するまでに時間差がないことがあります。この機能により受け入れ可能な接続が操作中にドロップされる可能性が削減されます。



**ヒント** ルールタイプのトランザクションモデルをイネーブルにする場合、コンパイルの先頭と末尾をマークするSyslogが生成されます。これらのSyslogには780001～780004までの番号が付けられます。

ルールエンジンのトランザクションコミットモデルを有効にするには、次の手順を使用します。

#### 手順

ルールエンジンのトランザクションコミットモデルを有効にします。

**asp rule-engine transactional-commit option**

オプションは次のとおりです。

- **access-group** : グローバルにまたはインターフェイスに適用されるアクセスルール。

- **nat** : ネットワーク アドレス変換ルール。

例 :

```
ciscoasa(config)# asp rule-engine transactional-commit access-group
```

---

## ASP ロード バランシングの有効化

ASP のロード バランシング機能によって、次の問題を回避しやすくなります。

- フロー上での突発的なトラフィックの増加によって発生するオーバーラン
- 特定のインターフェイス受信リングをオーバーサブスクライブするバルク フローによるオーバーラン
- 比較的高過負荷のインターフェイス受信リングによるオーバーラン (シングルコアでは負荷を維持できません)

ASP ロードバランシングにより、1つのインターフェイス受信リングから受信したパケットを複数のコアが同時に処理できます。システムがパケットをドロップし、**show cpu** コマンドの出力が 100% を大きく下回る場合、互いに関連のない多数の接続にパケットが属しているのであれば、この機能によってスループットが向上することがあります。

手順

---

**ステップ 1** ASP ロード バランシングの自動オン/オフ切り替えを次のようにイネーブルにします。

**asp load-balance per-packet auto**

ASAv では **auto** キーワードを使用できません。手動で ASP ロード バランシングを有効化または無効化する必要があります。

**ステップ 2** 次のように手動で ASP ロード バランシングをイネーブルにします。

**asp load-balance per-packet**

ASP ロード バランシングは、**auto** コマンドを有効にしている場合でも、手動で無効化するまでは有効です。

**ステップ 3** 次のように ASP ロード バランシングを手動でディセーブルにします。

**no asp load-balance per-packet**

このコマンドは、手動で ASP ロード バランシングをイネーブルにした場合にのみ適用されます。**auto** コマンドも有効にしている場合、ASP ロード バランシングは自動的に有効または無効な状態に戻ります。

---

## DNS キャッシュのモニタリング

ASA では、特定のクライアントレス SSL VPN および `certificate` コマンドに送信された外部 DNS クエリーの DNS 情報のローカル キャッシュを提供します。各 DNS 変換要求は、ローカル キャッシュで最初に検索されます。ローカル キャッシュに情報がある場合、結果の IP アドレスが戻されます。ローカル キャッシュで要求を解決できない場合、設定されているさまざまな DNS サーバに DNS クエリーが送信されます。外部 DNS サーバによって要求が解決された場合、結果の IP アドレスが、対応するホスト名とともにローカル キャッシュに格納されます。

DNS キャッシュ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 dns-hosts`

DNS キャッシュを表示します。これには、DNS サーバからダイナミックに学習したエントリと `name` コマンドを使用して手動で入力された名前および IP アドレスが含まれます。

## 基本設定の履歴

機能名	プラットフォームリリース	説明
ISA 3000 ハードウェアバイパス	94(1225)	ISA 3000 は、トラフィックが電源喪失時にアプライアンスを通過し続けるようにするハードウェアバイパス機能をサポートします。  次のコマンドが導入されました。 <b>hardware-bypass</b> 、 <b>hardware-bypass manual</b> 、 <b>hardware-bypass boot-delay</b> 、 <b>show hardware-bypass</b>
自動 ASP ロードバランシング	9.3(2)	ASP ロードバランシング機能の自動切替を有効または無効に設定できるようになりました。  (注) 自動機能は ASA v ではサポートされません。手動による有効化または無効化のみがサポートされます。  次のコマンドが導入されました。 <b>asp load-balance per-packet-auto</b>

機能名	プラットフォームリリース	説明
デフォルトの Telnet パスワードの削除	9.0(2)、9.1(2)	<p>ASA への管理アクセスのセキュリティ向上のために、Telnet のデフォルト ログインパスワードが削除されました。Telnet を使用してログインする前に、パスワードを手動で設定する必要があります。</p> <p>(注) ログインパスワードが使用されるのは、Telnet ユーザ認証 (<b>aaa authentication telnet console</b> コマンド) を設定しない場合の Telnet に対してのみです。</p> <p>以前はパスワードをクリアすると、ASA がデフォルト「cisco」を復元していました。今ではパスワードをクリアすると、パスワードは削除されるようになりました。</p> <p>ログインパスワードは、スイッチから ASASM への Telnet セッションでも使用されず (<b>session</b> コマンドを参照)。最初 ASASM のアクセスでは、ログインパスワードを設定するまで、<b>service-module session</b> コマンドを使用します。</p> <p><b>password</b> コマンドが変更されました。</p>
パスワード暗号化の可視性	8.4(1)	<p><b>show password encryption</b> コマンドが変更されました。</p>
マスターパスフレーズ	8.3(1)	<p>この機能が導入されました。マスターパスフレーズを利用すると、プレーンテキストのパスワードが安全に、暗号化形式で保存され、1つのキーを使用してすべてのパスワードを一様に暗号化またはマスキングできるようになります。このようにしても、機能は一切変更されません。</p> <p>次のコマンドが導入されました。<b>key config-key password-encryption</b>、<b>password encryption aes</b>、<b>clear configure password encryption aes</b>、<b>show running-config password encryption aes</b>、<b>show password encryption</b></p>