



コマンドライン インターフェイスの使用

この章では、セキュリティ アプライアンスでの CLI の使用方法について説明します。この章は次の内容で構成されています。

- 「ファイアウォール モードとセキュリティ コンテキスト モード」 (P.7)
- 「コマンドのモードとプロンプト」 (P.8)
- 「構文の書式」 (P.9)
- 「コマンドの短縮形」 (P.9)
- 「コマンドラインの編集」 (P.9)
- 「コマンドの補完」 (P.10)
- 「コマンドのヘルプ」 (P.10)
- 「show コマンド出力のフィルタリング」 (P.10)
- 「コマンド出力のページング」 (P.12)
- 「コメントの追加」 (P.13)
- 「テキスト コンフィギュレーション ファイル」 (P.13)



(注)

この CLI では構文など、Cisco IOS CLI と類似した表記法を使用しますが、セキュリティ アプライアンスのオペレーティング システムが Cisco IOS ソフトウェアのいずれかのバージョンに該当するわけではありません。Cisco IOS CLI コマンドがセキュリティ アプライアンスで動作するわけでも、同じ機能を使用できるわけでもありませんので注意してください。

ファイアウォール モードとセキュリティ コンテキスト モード

セキュリティ アプライアンスは、次のモードの組み合わせで動作します。

- トランスペアレント ファイアウォール モードまたはルーテッド ファイアウォール モード
ファイアウォール モードは、セキュリティ アプライアンスがレイヤ 2 ファイアウォールまたはレイヤ 3 ファイアウォールのいずれとして動作するかを決定します。
- マルチ コンテキスト モードまたはシングル コンテキスト モード

セキュリティ コンテキスト モードは、セキュリティ アプライアンスが単一のデバイスとして動作するか、またはマルチセキュリティ コンテキストとして動作する（仮想デバイスのように動作する）かを決定します。

特定のモードでしか使用できないコマンドもあります。

コマンドのモードとプロンプト

セキュリティ アプライアンスの CLI にはコマンド モードが含まれています。特定のモードでしか入力できないコマンドもあります。たとえば、機密情報を表示するコマンドを入力するには、パスワードを入力して特権モードに入る必要があります。次に、コンフィギュレーション変更が誤って入力されないようにするために、コンフィギュレーション モードに入る必要があります。下位のコマンドはすべて、高位のモードで入力できます。たとえば、グローバル コンフィギュレーション モードで特権 EXEC コマンドを入力することができます。

システム コンフィギュレーション モードまたはシングル コンテキスト モードに入っている場合、プロンプトはホスト名で始まります。

```
hostname
```

コンテキスト内に入っている場合、プロンプトはホスト名で始まり、その後にコンテキスト名が続きます。

```
hostname/context
```

プロンプトは、アクセス モードに応じて変化します。

- ユーザ EXEC モード

ユーザ EXEC モードでは、最小限のセキュリティ アプライアンス 設定が表示されます。ユーザ EXEC モードのプロンプトは、初めてセキュリティ アプライアンス にアクセスしたときに次のように表示されます。

```
hostname>
```

```
hostname/context>
```

- 特権 EXEC モード

特権 EXEC モードでは、ユーザの特権レベルまでの現在の設定がすべて表示されます。すべてのユーザ EXEC モード コマンドは、特権 EXEC モードで動作します。特権 EXEC モードを開始するには、ユーザ EXEC モードで **enable** コマンドを入力します。これにはパスワードが必要です。プロンプトにはシャープ記号 (#) が含まれています。

```
hostname#
```

```
hostname/context#
```

- グローバル コンフィギュレーション モード

グローバル コンフィギュレーション モードでは、セキュリティ アプライアンス コンフィギュレーションを変更できます。このモードでは、ユーザ EXEC、特権 EXEC、およびグローバルの各コンフィギュレーション コマンドをすべて使用できます。グローバル コンフィギュレーション モードを開始するには、特権 EXEC モードで **configure terminal** コマンドを入力します。プロンプトが次のように変化します。

```
hostname (config) #
```

```
hostname/context (config) #
```

- コマンド固有のコンフィギュレーション モード

いくつかのコマンドは、グローバル コンフィギュレーション モードから、コマンド固有のコンフィギュレーション モードに移行します。このモードでは、ユーザ EXEC、特権 EXEC、グローバルの各コンフィギュレーション コマンド、およびコマンド固有のコンフィギュレーション コマンドをすべて使用できます。たとえば、**interface** コマンドを使用すると、インターフェイス コンフィギュレーション モードに入ります。プロンプトが次のように変化します。

```
hostname (config-if) #
hostname/context (config-if) #
```

構文の書式

コマンド構文の説明には、次の表記法を使用しています。

表 1 構文の表記法

表記法	説明
太字	記載されているとおりに入力するコマンドおよびキーワードは、太字で示しています。
イタリック体	イタリック体の文字は、ユーザが値を指定する引数です。
[x]	省略可能な要素（キーワードまたは引数）は、角カッコで囲んで示しています。
	省略可能または必須のキーワードや引数の中から選択する場合は、縦棒で区切って示しています。
[x y]	いずれか 1 つを選択できる省略可能なキーワードや引数は、角カッコで囲み、縦棒で区切って示しています。
{x y}	必ずいずれか 1 つを選択しなければならない必須キーワードや引数は、波カッコで囲み、縦棒で区切って示しています。
[x {y z}]	省略可能または必須の要素内に、さらに省略可能または必須の選択肢を含める場合は、角カッコや波カッコを入れ子にして示しています。角カッコ内の波カッコと縦棒は、省略可能な要素内で選択すべき必須の要素を示しています。

コマンドの短縮形

ほとんどのコマンドは、コマンドに固有の最小文字数まで短縮できます。たとえば、コンフィギュレーションを表示するには、完全なコマンド **write terminal** を入力する代わりに、**wr t** と入力できます。または、特権モードを開始するには **en**、コンフィギュレーション モードを開始するには **conf t** と入力できます。さらに、**0** を入力して、**0.0.0.0** を表すことができます。

コマンドラインの編集

セキュリティ アプライアンスでは、Cisco IOS ソフトウェアと同じコマンドライン編集ルールが使用されます。**show history** コマンドを使用して以前入力した全コマンドを表示することも、↑キーまたは ^p コマンドで 1 つずつ前のコマンドを表示することもできます。前に入力したコマンドを確認したら、

↓キーまたは ^n コマンドでリスト内で前に進むことができます。再利用するコマンドに到達したら、そのコマンドを編集することも、**Enter** キーを押して実行することもできます。^w でカーソルの左側にある単語を削除することも、^u でカーソルのある行を消去することもできます。

セキュリティ アプライアンスでは、1 つのコマンドに 512 文字まで入力できます。512 文字を超えて入力した文字は無視されます。

コマンドの補完

部分的な文字列を入力してからコマンドまたはキーワードを完成させるには、**Tab** キーを押します。セキュリティ アプライアンスは、部分的な文字列がコマンドまたはキーワード 1 つだけと一致する場合に限り、コマンドまたはキーワードを完成させます。たとえば、**s** と入力して **Tab** キーを押した場合は、一致するコマンドが複数あるため、セキュリティ アプライアンスはコマンドを完成させません。ただし、**dis** を入力して **Tab** キーを押すと、コマンド **disable** が補完されます。

コマンドのヘルプ

次のコマンドを入力すると、コマンドラインからヘルプ情報を利用できます。

- **help *command_name***
特定のコマンドのヘルプを表示します。
- **help ?**
ヘルプがあるコマンドを表示します。
- ***command_name* ?**
使用可能な引数のリストを表示します。
- ***string*?** (スペースなし)
その文字列で始まるコマンドをリストします。
- **?および +?**
使用できるすべてのコマンドをリストします。**?** と入力すると、セキュリティ アプライアンスは現在のモードで使用できるコマンドだけを表示します。下位モードのコマンドも含め、使用できるすべてのコマンドを表示するには、**+?** と入力します。



(注) コマンド スtring に疑問符 (?) を含める場合は、不用意に CLI ヘルプが起動しないよう、疑問符を入力する前に **Ctrl+V** を押す必要があります。

show コマンド出力のフィルタリング

縦棒 (|) はどの **show** コマンドでも使用できます。これには、フィルタ オプションとフィルタリング式を組み込むことができます。フィルタリングは、Cisco IOS ソフトウェアと同様に、各出力行を正規表現と照合することによって行われます。選択するフィルタ オプションによって、正規表現に一致するすべての出力を含めたり除外したりできます。また、正規表現に一致する行で始まるすべての出力を表示することもできます。

show コマンドでフィルタリング オプションを使用する場合の構文は、次のとおりです。

```
hostname# show command | {include | exclude | begin | grep [-v]} regexp
```

このコマンド文字列の最初の縦棒 (|) は演算子であり、コマンド内に含める必要があります。この演算子は、**show** コマンドの出力をフィルタに誘導します。構文内に含まれるその他の縦棒 (|) は代替オプションを示すものであり、コマンドの一部ではありません。

include オプションを指定すると、正規表現に一致するすべての出力行が表示されます。**-v** を付けずに **grep** オプションを使用する場合も、同じ結果となります。**exclude** オプションを指定すると、正規表現に一致するすべての出力行が除外されます。**-v** を付けて **grep** オプションを使用する場合も、同じ結果となります。**begin** オプションを指定すると、正規表現に一致する行で始まるすべての出力行が表示されます。

regexp には、Cisco IOS の正規表現を指定します。正規表現は一重引用符または二重引用符で囲まれていません。したがって、末尾の空白スペースが正規表現の一部と解釈されるため、末尾の空白スペースに注意してください。

正規表現を作成する場合は、照合する任意の文字または数字を使用できます。また、メタ文字と呼ばれるキーボード文字は、正規表現で使用されると特別な意味を持ちます。

Ctrl キーを押した状態で **V** キーを押すと、CLI において、疑問符 (?) やタブなどの特殊文字をすべてエスケープできます。たとえば、コンフィギュレーションで **d?g** と入力するには、**d[Ctrl+V]?g** とキー入力します。

表 2 は、特殊な意味を持つメタ文字のリストです。

表 2 regex メタ文字

文字	説明	注意事項
.	ドット	任意の単一文字と一致します。たとえば、 d.g は、 dog 、 dag 、 dtg 、およびこれらの文字を含む任意の単語 (doggonnit など) に一致します。
(exp)	サブ表現	サブ表現は、文字を周囲の文字から分離して、サブ表現に他のメタ文字を使用できるようにします。たとえば、 d(o)a g は dog および dag に一致しますが、 do a g は do および ag に一致します。また、サブ表現を繰り返し限定作用素とともに使用して、繰り返す文字を区別できます。たとえば、 ab(xy){3}z は、 abxyxyxyz に一致します。
	代替	このメタ文字によって区切られている複数の表現のいずれかと一致します。たとえば、 dog cat は、 dog または cat に一致します。
?	疑問符	直前の表現が 0 または 1 個存在することを示す修飾子。たとえば、 lo?se は、 lse または lose に一致します。 (注) Ctrl+V を入力してから疑問符を入力しないと、ヘルプ機能が呼び出されます。
*	アスタリスク	直前の表現が 0、1、または任意の個数存在することを示す修飾子。たとえば、 lo*se は、 lse 、 lose 、 loose などに一致します。
+	プラス	直前の表現が少なくとも 1 個存在することを示す修飾子。たとえば、 lo+se は、 lose および loose に一致しますが、 lse には一致しません。

表 2 regex メタ文字 (続き)

文字	説明	注意事項
{x} または {x,}	最小繰り返し限定作用素	少なくとも x 回繰り返します。たとえば、 ab(xy){2,}z は、 abxyxyz や abxyxyxyz などに一致します。
[abc]	文字クラス	カッコ内の任意の文字と一致します。たとえば、 [abc] は、 a 、 b 、または c に一致します。
[^abc]	否定文字クラス	角カッコに含まれていない単一文字と一致します。たとえば、 [^abc] は、 a 、 b 、 c 以外の任意の文字に一致します。 [^A-Z] は、大文字のアルファベット文字以外の任意の単一の文字に一致します。
[a-c]	文字範囲クラス	範囲内の任意の文字と一致します。 [a-z] は、任意の小文字のアルファベット文字に一致します。文字と範囲を組み合わせることもできます。 [abcq-z] および [a-cq-z] は、 a 、 b 、 c 、 q 、 r 、 s 、 t 、 u 、 v 、 w 、 x 、 y 、 z に一致します。 ダッシュ (-) 文字は、角カッコ内の最初の文字または最後の文字である場合にのみリテラルとなります ([abc-] や [-abc])。
""	引用符	文字列の末尾または先頭のスペースを保持します。たとえば、 " test" は、一致を検索する場合に先頭のスペースを保持します。
^	キャレット	行の先頭を指定します。
\	エスケープ文字	メタ文字とともに使用すると、リテラル文字と一致します。たとえば、 \ は左角カッコに一致します。
<i>char</i>	文字	文字がメタ文字でない場合は、リテラル文字と一致します。
\r	復帰	復帰 0x0d と一致します。
\n	改行	改行 0x0a と一致します。
\t	Tab	タブ 0x09 と一致します。
\f	改ページ	フォーム フィード 0x0c と一致します。
\xNN	エスケープされた 16 進数	16 進数 (厳密に 2 桁) を使用した ASCII 文字と一致します。
\NNN	エスケープされた 8 進数	8 進数 (厳密に 3 桁) としての ASCII 文字と一致します。たとえば、文字 040 はスペースを表します。

コマンド出力のページング

help、**?**、**show**、**show xlate** や、リストが長いその他のコマンドなどでは、画面に情報を表示して停止するか、完了するまでコマンドを実行させるかを指定できます。**pager** コマンドを使用すると、画面に表示する行数を選択して、その行数を表示した後に **More** プロンプトを表示することができます。

ページングがイネーブルになっているときには、次のプロンプトが表示されます。

```
<--- More --->
```

More プロンプトの構文は、UNIX の **more** コマンドと似ています。

- 次の 1 画面分の情報を表示するには、スペース バーを押します。

- 次の行を表示するには、Enter キーを押します。
- コマンドラインに戻るには、q キーを押します。

コメントの追加

行の先頭にコロン (:) を置いて、コメントを作成できます。しかし、コメントが表示されるのはコマンド ヒストリ バッファだけで、コンフィギュレーションには表示されません。したがって、コメントは、**show history** コマンドを使用するか、矢印キーを押して前のコマンドを取得することによって表示できますが、コンフィギュレーションには含まれないので、**write terminal** コマンドでは表示できません。

テキスト コンフィギュレーション ファイル

この項では、セキュリティ アプライアンスにダウンロードできるテキスト コンフィギュレーション ファイルをフォーマットする方法について説明します。次の項目を取り上げます。

- 「テキスト ファイルでコマンドと行が対応する仕組み」(P.13)
- 「コマンド固有のコンフィギュレーション モード コマンド」(P.13)
- 「自動テキスト入力」(P.14)
- 「行の順序」(P.14)
- 「テキスト コンフィギュレーションに含まれないコマンド」(P.14)
- 「パスワード」(P.14)
- 「マルチセキュリティ コンテキスト ファイル」(P.14)

テキスト ファイルでコマンドと行が対応する仕組み

テキスト コンフィギュレーション ファイルには、このガイドで説明するコマンドに対応する行が含まれています。

例では、コマンドの前に CLI プロンプトがあります。次の例では、CLI プロンプトは「hostname(config)#」です。

```
hostname (config)# context a
```

テキスト コンフィギュレーション ファイルでは、コマンドの入力を求めるプロンプトが表示されないため、プロンプトは省略されています。

```
context a
```

コマンド固有のコンフィギュレーション モード コマンド

コマンド固有のコンフィギュレーション モード コマンドは、コマンドラインで入力されたときに、メイン コマンドの下に字下げして表示されます。テキスト ファイルの行は、コマンドがメイン コマンドのすぐ後に表示される限り、字下げする必要はありません。たとえば、次のテキストは字下げされていませんが、字下げしたテキストと同じように読み取られます。

```
interface gigabitethernet0/0
```

```
nameif inside
interface gigabitethernet0/1
    nameif outside
```

自動テキスト入力

コンフィギュレーションをセキュリティ アプライアンスにダウンロードすると、セキュリティ アプライアンスにより一部の行が自動的に挿入されます。たとえば、セキュリティ アプライアンスは、デフォルト設定のため、またはコンフィギュレーションが変更されたときのための行を挿入します。テキスト ファイルを作成するときは、これらの自動入力を行う必要はありません。

行の順序

ほとんどの場合、コマンドはファイル内で任意の順序に置くことができます。ただし、ACE などいくつかの行は表示された順に処理されるので、順序がアクセス リストの機能に影響する場合があります。その他のコマンドでも、順序の要件がある場合があります。たとえば、あるインターフェイスの名前を多数の後続コマンドが使用する場合は、そのインターフェイスの **nameif** コマンドをまず入力する必要があります。また、コマンド固有のコンフィギュレーション モードのコマンドは、メイン コマンドの直後に置く必要があります。

テキスト コンフィギュレーションに含まれないコマンド

いくつかのコマンドは、コンフィギュレーションに行を挿入しません。たとえば、**show running-config** などのランタイム コマンドは、テキスト ファイル内に対応する行があります。

パスワード

ログイン パスワード、イネーブル パスワード、およびユーザ パスワードは、コンフィギュレーションに保存される前に自動的に暗号化されます。たとえば、パスワード「cisco」の暗号化された形式は jMorNbK0514fadBh のようになります。コンフィギュレーション パスワードは暗号化された形式で別のセキュリティ アプライアンスにコピーできますが、そのパスワードの暗号を解読することはできません。

暗号化されていないパスワードをテキスト ファイルに入力した場合、コンフィギュレーションをセキュリティ アプライアンスにコピーしても、セキュリティ アプライアンスは自動的にパスワードを暗号化しません。セキュリティ アプライアンスがパスワードを暗号化するのは、**copy running-config startup-config** コマンドまたは **write memory** コマンドを使用して、コマンドラインから実行コンフィギュレーションを保存した場合のみです。

マルチセキュリティ コンテキスト ファイル

マルチ セキュリティ コンテキストの場合、コンフィギュレーション全体は次に示す複数の部分で構成されます。

- セキュリティ コンテキスト コンフィギュレーション
- コンテキストのリストなど、セキュリティ アプライアンスの基本設定を示すシステム コンフィギュレーション

- システム コンフィギュレーション用のネットワーク インターフェイスを提供する管理コンテキスト

システム コンフィギュレーションには、それ自体のインターフェイスまたはネットワーク設定は含まれていません。代わりに、システムは、ネットワーク リソースにアクセスする必要があるときに（サーバからコンテキストをダウンロードするときなど）、管理コンテキストとして指定されたコンテキストを使用します。

各コンテキストは、シングル コンテキスト モード コンフィギュレーションに似ています。システム コンフィギュレーションにはシステム限定のコマンド（全コンテキストのリストなど）が含まれており、その他の一般的なコマンド（多数のインターフェイス パラメータなど）は存在しない点で、システム コンフィギュレーションは、コンテキスト コンフィギュレーションとは異なっています。

