



# はじめに

---

この章は、次の項で構成されています。

- [概要 \(2 ページ\)](#)
- [ユーザ特権レベル \(3 ページ\)](#)
- [CLI コマンドモード \(5 ページ\)](#)
- [デバッグアクセス用のインターフェイス \(8 ページ\)](#)
- [CLI のアクセス \(9 ページ\)](#)
- [CLI コマンドの表記法 \(11 ページ\)](#)
- [機能の編集 \(12 ページ\)](#)
- [インターフェイス命名規則 \(15 ページ\)](#)
- [IPv6z アドレスの表記法 \(17 ページ\)](#)
- [ループバック インターフェイス \(18 ページ\)](#)
- [CLI によるポートの管理 \(20 ページ\)](#)
- [リモート IP アドレスと OOB ポート \(21 ページ\)](#)
- [PHY 診断 \(22 ページ\)](#)
- [CLI 出力修飾子 \(23 ページ\)](#)

## 概要

CLIはさまざまなコマンドモードに分けられます。各モードには、コマンドのグループが含まれます。

これらのモードについては、[CLI コマンドモード \(5 ページ\)](#) で説明します。

ユーザには、特権レベルが割り当てられます。各ユーザ権限レベルで特定のCLIモードにアクセスできます。

次の項では、ユーザ レベルについて説明します。

## ユーザ特権レベル

ユーザは、次のいずれかのユーザ レベルを使用して作成できます。

- レベル1：このレベルのユーザは、ユーザ EXEC モード コマンドのみを実行できます。このレベルのユーザは、web GUI またはコマンドに特権 EXEC モードでアクセスできません。
- レベル7：このレベルのユーザは、コマンドをユーザ EXEC モードで実行したり、コマンドのサブセットを特権 EXEC モードで実行したりできます。このレベルのユーザは web GUI にアクセスできません。
- レベル15：このレベルのユーザはすべてのコマンドを実行できます。このレベルのユーザのみが web GUI にアクセスできます。

システム管理者（レベル 15 のユーザ）は、低レベルのユーザが高レベルのユーザに一時的に昇格できるパスワードを作成できます。たとえば、ユーザのレベルを 1 から 7 に、1 から 15 に、7 から 15 などに昇格できます。

各レベルのパスワードは、次のコマンドを使用して（管理者が）設定します。

```
enable password [level privilege-level] {password|encrypted encrypted-password}
```

このパスワードを使用すると、**enable** コマンドとレベル 7 または 15 のパスワードを入力してユーザ レベルを昇格できます。レベル 1 からレベル 7 に、またはレベル 15 に直接昇格できます。高レベルは現在のセッションでのみ保持されます。

**disable** コマンドにより、ユーザは低レベルに戻されます。

ユーザを作成してユーザ レベルを割り当てるには、**username** コマンドを使用します。このレベルのユーザを作成できるのはコマンド レベル 15 のユーザのみです。

例：（管理者が）レベル 7 および 15 のパスワードを作成します。

```
switchxxxxxx#configure
switchxxxxxx<conf># enable password level 7 level7@aBc
switchxxxxxx<conf># enable password level 15 level15@aBc
switchxxxxxx<conf>#
```

ユーザ レベル 1 のユーザを作成します。

```
switchxxxxxx#configure
switchxxxxxx<conf> username john password John1234 privilege 1
switchxxxxxx<conf>
```

例 2：レベル 1 とレベル 15 を切り替えます。ユーザにはパスワードが必要です。

```
switchxxxxxx#
switchxxxxxx# enable
Enter Password: ***** (this is the password for level 15
- Level15@abc)
switchxxxxxx#
```



---

(注) パスワードの認証が RADIUS または TACACS+ サーバで実行される場合、ユーザレベル 7 およびユーザレベル 15 に割り当てるパスワードは外部サーバで設定し、\$enable7\$ と \$enable15\$ のユーザ名に個別に関連付ける必要があります。

---

# CLI コマンドモード

CLIは4つのコマンドモードに分けられます。コマンドモードは次のとおりです（アクセス順）。

- ユーザ EXEC モード
- 特権 EXEC モード
- グローバル コンフィギュレーション モード

各コマンドモードには、独自の固有なコンソールプロンプトおよびCLIコマンドセットがあります。コンソールプロンプトで疑問符を入力すると、現在のモードとユーザのレベルで利用可能なコマンドのリストが表示されます。特定のコマンドは、モードを切り替えるために使用します。

ユーザには、モードとそこで利用可能なコマンドを決定する権限レベルが割り当てられます。

## ユーザ EXEC モード

レベル1のユーザは、最初にユーザEXECモードにログインします。ユーザEXECモードは、基本的なテストの実行やシステム情報の表示などの設定を変更しないタスクで使用されます。

ユーザレベルプロンプトでは、スイッチホスト名の後に#が続きます。デフォルトホスト名はswitchxxxxxxで、xxxxxxは次に示すようにデバイスのMACアドレスの最後の6桁を示します

```
switchxxxxxx#
```

デフォルトのホスト名は、hostnameコマンドを介してグローバルコンフィギュレーションモードで変更できます。

## 特権 EXEC モード

レベル7または15のユーザは特権EXECモードに自動的にログインします。

レベル1のユーザは、enableコマンドを入力してプロンプトが表示されたらレベル15のパスワードを入力すると、特権EXECモードを開始できます。

特権EXECモードからユーザEXECモードに戻るには、desableコマンドを使用します。

## 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を使用すると、インターフェイスレベルではなく、システムレベルで機能を設定するコマンドを実行できます。

コマンドレベル7または15のユーザだけがこのモードでアクセスできます。

グローバルコンフィギュレーションモードを特権EXECモードからアクセスするには、configureコマンドを特権EXECモードプロンプトで入力してEnterを押します。グローバルコンフィ

ギューレーションモードプロンプトには、デバイスホスト名の後に (config)# が続けて表示されます。

```
switchxxxxx(config)#
```

グローバルコンフィギュレーションモードから特権 EXEC モードに戻るには、次のいずれかのコマンドを使用します。

- exit
- end
- Ctrl+Z

次の例では、グローバルコンフィギュレーションモードにアクセスして特権 EXEC モードに戻る方法を示します。

```
switchxxxxx#  
switchxxxxx# configure  
switchxxxxx(config)# exit  
switchxxxxx#
```

### インターフェイスまたは回線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からさまざまなサブモードを入力できます。これらのサブモードは、

インターフェイスまたは回線のグループでコマンドを実行できるようにします。

たとえば、特定のポートまたはポートの範囲でいくつかの操作を実行する場合は、

そのインターフェイスのインターフェイスコンフィギュレーションモードを開始できます。

次に、vlan1 でインターフェイスコンフィギュレーションモードを開始し、

速度を設定する例を示します。

グローバルコンフィギュレーションモードに戻るには exit コマンドを使用します。

```
switchxxxxx#  
switchxxxxx# configure  
switchxxxxx(config)# interface range vlan1  
switchxxxxx(config-if)# speed 10  
switchxxxxx(config-if)# exit  
switchxxxxx(config)#
```

次に、使用可能ないくつかのサブモードの例を示します。

- インターフェイス：特定のインターフェイス（ポート、VLAN、ポートチャネル、またはトンネル）またはインターフェイス範囲を設定するコマンドが含まれます。グローバルコンフィギュレーションモードコマンド `interface` 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を開始できます。グローバルコンフィギュレーションコマンド `interface` を使用すると、このモードを開始できます。
- 回線インターフェイス：コンソール、Telnet、SSH の管理接続の設定に使用するコマンドが含まれます。回線タイムアウト設定などのコマンドが含まれます。グローバルコンフィギュレーションコマンド `line` を使用すると、回線設定コマンドモードを開始できます。

- **VLAN データベース**：VLAN 全体の設定に使用するコマンドが含まれます。vlan database グローバル コンフィギュレーション モード コマンドを使用すると、VLAN データベース インターフェイス コンフィギュレーション モードを開始できます。
- **管理アクセス リスト**：管理アクセス リストの定義に使用するコマンドが含まれます。management access-list グローバル コンフィギュレーション モード コマンドを使用すると、管理アクセス リスト コンフィギュレーション モードを開始できます。
- **MAC アクセスリスト、IPv6 アクセスリスト、IP アクセスリスト**：MAC アドレス、IPv6 アドレス、および IPv4 アドレスのそれぞれに基づいてトラフィックを許可するために必要な条件を設定します。これらのコンフィギュレーション モードを開始するには、mac access-list、ipv6 access-list、および ip access-list グローバル コンフィギュレーション モード コマンドを使用します。

インターフェイス コンフィギュレーション モードからグローバル コンフィギュレーション モードに戻るには、exit コマンドを使用します。

## デバッグアクセス用のインターフェイス

上述の標準CLIインターフェイスモードに加えて、デバイスはデバイスデバッグアクセス用の追加インターフェイスをサポートしています。これらのインターフェイスは、デバイスの動作をデバッグする必要がある場合に、シスコサポートチームの担当者が使用することを目的としています。これらのインターフェイスはパスワードで保護されています。

デバイスは、次のデバッグインターフェイスをサポートしています。

- ブートシーケンス時のU-BOOTアクセス（シリアルコンソール端末経由でのみアクセス可能）
- ブートシーケンス時のLinuxカーネルアクセス（シリアルコンソール端末からのみアクセス可能）
- 実行時デバッグモード：シスコサポートチームの担当者がデバイス設定を表示し、プロトコルとレイヤ1のデバッグコマンドと設定を適用できます（シリアル、Telnet、またはSSHコンソール経由でアクセス可能）

これらのインターフェイスのパスワードは、次のように生成されます。

- デバッグインターフェイスにアクセスすると、デバイスはランダムなハッシュ値を生成し、画面に表示します。
- ハッシュ値は、秘密キーを使用してセキュアなシスコサーバーで署名するために、デバイス管理者とシスコのサポート担当者によって送信されます。
- この操作の出力は、デバッグインターフェイスアクセスのパスワードとして使用されます。
- このパスワードは現在のセッションに適しています。次回デバッグインターフェイスに入ろうとする時、またはデバイスの再起動後には、デバイスは新しいランダムハッシュを生成します。



# CLI のアクセス

CLIには、次のタスクのいずれかを実行して端末またはコンピュータからアクセスできます。

- HyperTerminal などの端末アプリケーションをスイッチのコンソールポートに直接接続されているコンピュータの COM ポートで実行するか、  
または
- スイッチとネットワークで接続されたコンピュータでコマンドプロンプトから Telnet セッションを実行する。
- スイッチへのネットワーク接続があるコンピュータで実行している SSH クライアントをサポートするアプリケーションから SSH を使用する。



(注) デフォルトでは、スイッチの Telnet および SSH は無効です。

Telnet 接続または SSH 接続でアクセスする場合、CLI コマンドを使用する前に次の条件を満たしていることを確認します。

- スイッチには IP アドレスが定義されている
- 対応する管理アクセスが有効になっている
- コンピュータとスイッチが相互に接続できるように IP パスがある

## コンソールインターフェイスを介した端末の使用

デバイスは、デュアルコンソール管理インターフェイス (Type-C USB インターフェイスと RJ45 ポート) をサポートしています。Type-C USB と RJ45 の両方が接続されている場合は、Type-C USB インターフェイスが優先されます。Type-C インターフェイスをサポートするために、管理ステーションへのドライバのインストールが必要になる場合があります。



(注) Type-C USB インターフェイスは、デバイスの電源がオン/リブートされてから数秒後にアクティブになります。

コンピュータとスイッチを接続したら、CLI にアクセスするための端末アプリケーションを実行します。ターミナルエミュレータは、`databits=8` と `parity=none` になるように設定する必要があります。

[Enter] を 2 回クリックし、デバイスで PC のシリアルポート速度に対応するシリアルポート速度を設定します。

CLI が表示されたら、[User Name] プロンプトに `cisco` と入力し、[Password] プロンプトに `cisco` と入力します。



- (注) デフォルトのユーザ名とパスワードを使用して初めてログインすると、デバイスにはユーザ名とパスワードを変更するプロンプトが表示されます。新しいパスワードは、パスワードの複雑さのルールを順守する必要があります。

switchxxxxx# のプロンプトが表示されます。

CLI コマンドを入力してスイッチを管理できるようになりました。CLI コマンドの詳細については、このリファレンス ガイドの該当する章を参照してください。

## イーサネット インターフェイス上で Telnet を使用する

Telnet は、IP ネットワークを介して CLI に接続する方法を提供します。

コマンドプロンプトから telnet セッションを確立するには、次の手順を実行します。

- ステップ 1 [Start] をクリックし、[All Programs] > [Accessories] > [Command Prompt] を選択してコマンドプロンプトを開きます。
- ステップ 2 プロンプトに **telnet 1<IP address of switch>** と入力し、[Enter] を押します。
- ステップ 3 CLI が表示されます。
- ステップ 4 CLI が表示されたら、[UserName] プロンプトで定義したユーザ名を入力し、定義したパスワードを [Password] プロンプトに入力します。

- (注) デフォルトのユーザ名とパスワードを使用して初めてログインすると、デバイスにはユーザ名とパスワードを変更するプロンプトが表示されます。新しいパスワードは、パスワードの複雑さのルールを順守する必要があります。

switchxxxxx# のプロンプトが表示されます。CLI コマンドを入力してスイッチを管理できるようになりました。CLI コマンドの詳細については、このリファレンス ガイドの該当する章を参照してください。

## CLI コマンドの表記法

コマンドを入力する場合、すべてのコマンドに適用される特定のコマンド入力標準があります。次の表では、コマンド表記法について説明します。

表記法	説明
[ ]	コマンドラインで、角カッコはオプション入力のことを示します。
{ }	コマンドラインで、中カッコは必須パラメータを区切る   文字の選択範囲を示します。オプションを1つ選択する必要があります。たとえば、 <b>flowcontrol {auto on off}</b> は flowcontrol コマンドを指し、auto、on、または off のいずれかを選択する必要があります。
"" (反転カンマ)	入力文字列にスペースや予約語 (つまり VLAN) が含まれている場合、文字列を反転カンマ内に配置します。
parameter	斜体はパラメータを示します。
押すキー	押すキーの名前は太字で表示されます。
Ctrl+F4	+ 文字で区切られたキーはキーボードで同時に押します
画面表示	固定長フォントは、CLI プロンプト、ユーザが入力した CLI コマンド、およびコンソールに表示されるシステム メッセージです。
all	ポートまたはパラメータの範囲の定義でパラメータが必要で、all がオプションにある場合、パラメータが定義されていないと、コマンドのデフォルト値は all になります。たとえば、 <b>interface range port-channel</b> コマンドでは、チャンネルの範囲を入力するオプションまたは all を選択するオプションのいずれかを指定します。パラメータを指定せずにコマンドを入力すると、デフォルト値は自動的に all になります。
text	テキストが空白で区切られた複数の文字で構成される場合 (snmp-server contact コマンドの場合など)、コマンドのパラメータとしてテキストを自由に入力するには、文字列全体を二重引用符で囲んで表示する必要があります。例: <b>snmp-server contact "QA on floor 8"</b>

## 機能の編集

### コマンドの入力

CLI コマンドは一連のキーワードと引数で構成されます。キーワードはコマンドを特定し、引数は設定パラメータを指定します。たとえば、`show interfaces status Gigabitethernet 1` コマンドでは、`show`、`interfaces`、および `status` はキーワードで、`Gigabitethernet` はインターフェイスタイプを指定する引数、`1` はポートを指定します。

パラメータが必要なコマンドを入力するには、コマンドキーワードの後に必要なパラメータを入力します。たとえば、管理者のパスワードを設定するには次のように入力します。

```
switchxxxxx(config)# username admin password Alansmith1
```

CLI を使用する場合、コマンドオプションは表示されません。ヘルプを要求するための標準コマンドは `?` です。

ヘルプ情報が表示される 2 つのインスタンスがあります。

- キーワードルックアップ：`?` 文字をコマンドの代わりに入力します。すべての有効なコマンドと対応するヘルプメッセージのリストが表示されます。
- 部分的なキーワードルックアップ：コマンドが不完全な場合にパラメータの代わりに `?` 文字を入力すると、このコマンドに一致するキーワードまたはパラメータが表示されます。

### 端末のコマンドバッファ

CLI でコマンドを入力するたびに、内部的に管理されているコマンド履歴バッファに記録されます。バッファに記録されているコマンドは先入れ先出し (FIFO) で保持されます。このコマンドは、呼び出し、確認、変更、および再発行を行うことができます。このバッファは、デバイスがリセットされると保持されません。

キーワード	説明
↑キー	履歴バッファ内のコマンドを呼び出します。最後に実行したコマンドが最初に呼び出されます。キーを押すたびに、より古いコマンドが順次表示されます。
↓キー	↑キーでコマンドを呼び出した後で、履歴バッファ内のより新しいコマンドに戻ります。キーを押すたびに、より新しいコマンドが順次表示されます。

デフォルトでは、履歴バッファシステムは有効ですが、いつでも無効にすることができます。履歴バッファの有効と無効の切り替えに関する詳細については、`history` コマンドを参照してください。

デフォルトでは、バッファには標準的な数のコマンドが保存されています。標準的な 10 個のコマンドを 216 個に増やすことができます。0 に設定すると、履歴バッファシステムを無効に

した場合と同じ効果が得られます。コマンド履歴バッファの設定に関する詳細については、**history size** コマンドを参照してください。

履歴バッファを表示する場合は、**show history** コマンドを参照してください。

### コマンドの影響を無効にする

多くの設定コマンドでは、プレフィックス キーワード **no** を入力すると、コマンドの影響を取り消したり、デフォルト値に対する設定をリセットしたりできます。このリファレンスガイドでは、各 CLI コマンドの無効効果について説明します。

### コマンドの補完

入力したコマンドが不完全な場合、無効な場合、パラメータが欠けているまたは無効な場合、適切なエラーメッセージが表示されます。このため、正しいコマンドを入力できます。不完全なコマンドを入力した後に **Tab** を押すと、コマンドを特定して完全なものにしようとします。すでに入力した文字が足りずに、システムが一致するコマンドを1つも特定できない場合は、**?** を押すと、すでに入力した文字と一致する利用可能なコマンドが表示されます。

### キーボードのショートカット

CLI には、CLI コマンドの編集に役立つ一連のキーボードショートカットが指定されています。次の表では、CLI ショートカットについて説明します。

キーボードのキー	説明
↑	履歴バッファからコマンドを呼び出します。最後に実行したコマンドが最初に呼び出されます。キーを押すたびに、より古いコマンドが順次表示されます。
↓	↑キーでコマンドを呼び出した後で、履歴バッファ内の最新のコマンドに戻ります。キーを押すたびに、より新しいコマンドが順次表示されます。
Ctrl+A	コマンドラインの先頭にカーソルを移動します。
Ctrl+E	カーソルをコマンドラインの末尾に移動します。
Ctrl+Z/End	コンフィギュレーションモードから特権 EXEC モードに戻ります。
Back Space	カーソル位置の左にある1つの文字を削除します。

### テキストのコピー アンド ペースト

デバイスには、最大 1000 行のテキスト（またはコマンド）をコピー アンド ペーストできません。



---

(注) ユーザの責任において、デバイスにコピーしたテキストが適切なコマンドのみで構成されるようにします。

---

設定ファイルからコマンドをコピーアンドペーストする場合は、次の条件を確認してください。

- デバイスのコンフィギュレーションモードにアクセスできる。

コマンドには、暗号化パスワードやキーなどの暗号化データを含めない。暗号化データの前に暗号化キーワードが使用される場合の暗号化パスワードを除いて、暗号化データをデバイスにコピーアンドペーストすることはできません (`enable password` コマンドの場合など)。

# インターフェイス命名規則

デバイスのインターフェイスは、次のタイプのいずれかにすることができます。

- ギガビットイーサネット（10/100/1000 キロビット）ポート：これらは GigabitEthernet、または gi、あるいは GE と記述されます。
- 2.5 ギガビットイーサネット（10/100/1000/25000 キロビット）ポート：これらは TwoPointFiveGigabitEthernet または tw と記述されます。
- 5 ギガビットイーサネット（10/100/1000/25000/50000 キロビット）ポート：これらは FiveGigabitEthernet または fi のいずれかで記述されます。
- LAG（ポートチャネル）：Port-Channel または po のいずれかで記述されます。
- VLAN：VLAN と記述されます。
- トンネル：tunnel または tu と記述されます。
- OOB：OutOfBand または oob と記述されます。

CLI で内では、インターフェイスは次の要素を連結して表されます。

- インターフェイスのタイプ：前述のとおり。
- ユニット番号：スタック内のユニット。
- スロット番号：スロット番号は常に 0 です。
- スタッキングモードでのインターフェイス名の構文は次のとおりです。  

```
{<port-type>[ ][<unit-number>]/<slot-number>/<port-number>} | {port-channel | po |
} [ <port-channel-number> |
{tunnel | tu} [ <tunnel-number> | vlan [ ]<vlan-id>
```
- インターフェイス番号：ポート、LAG、トンネル、または VLAN 番号。

次に、これらのさまざまなオプションの例を示します。

```
switchxxxxxx(config)#interface GigabitEthernet 1
switchxxxxxx(config)#interface GE 1
switchxxxxxx(config)#interface TwoPointFiveGigabitEthernet
switchxxxxxx(config)#interface po1
switchxxxxxx(config)# interface vlan 1
```

## インターフェイス範囲

インターフェイスは、個別にまたは範囲内で説明されています。インターフェイス範囲のコマンドは次のような構文になります。

```
<interface-range> ::=
{<port-type>[
```

```

]]<unit-number>/<slot-number>/<first-port-number>[ -
<last-port-number>] |
port-channel[ ]<first-port-channel-number>[ -
<last-port-channel-number>] |
tunnel[ ]<first-tunnel-number>[ - <last-tunnel-number>] |
vlan[ ]<first-vlan-id>[ - <last-vlan-id>]

```

このコマンドのサンプルを、次の例で示します。

```

switchxxxxxx#configure
switchxxxxxx(config-if)#interface range gil-5g

```

### 複数のインターフェイスタイプのリスト

インターフェイスタイプの組み合わせは、`interface range` コマンドで次の形式で指定できます。

```
<range-list> ::= <interface-range> | <range-list>, <interface-range>
```

最大 5 つの範囲を含めることができます。



- 
- (注) 範囲リストには、ポートとポートチャンネルまたは VLAN のいずれかを含められます。ポート/ポートチャンネルと VLAN の組み合わせは使用できません。
- 

カンマの後のスペースは省略可能です。

範囲リストを定義する場合、最初の入力後とカンマ (,) 前にスペースを入力する必要があります。

このコマンドのサンプルを、次の例で示します。

```

switchxxxxxx#configure
switchxxxxxx(config)#interface range gil-5, vlan 1-2

```



## IPv6z アドレスの表記法

次に、リンク ローカルの IPv6 アドレスである IPv6z アドレスを記述する方法について説明します。

形式 : <ipv6-link-local-address>%<egress-interface>

値は次のとおりです。

egress-interface (also known as zone) = vlan<vlan-id> | po<number> | tunnel<number> | port<number> | 0

出力インターフェイスが指定されていない場合、デフォルトのインターフェイスが選択されます。出力インターフェイス=0に指定することは、出力インターフェイスを定義しているわけではありません。

次の組み合わせを使用できます。

- ipv6\_address%egress-interface : 指定したインターフェイスの IPv6 アドレスを参照します。
- ipv6\_address%0 : IPv6 アドレスが定義される単一インターフェイスの IPv6 アドレスを参照します。
- ipv6\_address : IPv6 アドレスが定義される単一インターフェイスの IPv6 アドレスを参照します。

# ループバック インターフェイス

ルータ上の IP アプリケーションがリモート IP アプリケーションと通信する必要がある場合、その IP アドレスとして使用するローカル IP アドレスを選択する必要があります。ルータで定義された任意の IP アドレスを使用できますが、このリンクに障害が発生した場合、これらの IP アプリケーション間に別の IP ルートが用意されていても、通信が中断されます。

ループバック インターフェイスは仮想インターフェイスで、動作状態は常に稼働しています。この仮想インターフェイスで設定されている IP アドレスを、リモート IP アプリケーションと通信するときローカルアドレスとして使用する場合、リモート アプリケーションへの実際のルートが変更されていても、通信は中断されません。

ループバック インターフェイスの名前は `loopback1` です。

ループバック インターフェイスはブリッジをサポートしていません。いかなる VLAN のメンバーになることもできません。有効にできる レイヤ 2 プロトコルはありません。

## レイヤ 3 の指定

### IP インターフェイス

IPv4 および IPv6 アドレスはループバック インターフェイスに割り当てることができます。

IPv6 リンク ローカルのインターフェイス識別子は 1 です。

### ルーティング プロトコル

スイッチで実行されているルーティング プロトコルは、ルーティング プロトコルの再配布メカニズムを使用してループバック インターフェイスで定義された IP プレフィックスの通知をサポートしています。

## 設定例

### スタティック ルーティング

次の例で、スタティック ルーティングを使用するスイッチの IP を設定する方法を示します。

```
Switch# configure terminal
Switch(config)# interface vlan 1
Switch(config-if)# ip address 10.10.10.2 /24
Switch(config-if)# ipv6 address 2001:DB8:2222:7270::2312/64
Switch(config-if)# exit
Switch(config)# interface vlan 2
Switch(config-if)# ip address 10.11.11.2 /24
Switch(config-if)# ipv6 address 2001:DB8:3333:7271::2312/64
Switch(config-if)# exit
Switch(config)# interface loopback 1
Switch(config-if)# ip address 172.25.13.2 /32
Switch(config-if)# ipv6 address 2001:DB8:2222:7272::72/128
Switch(config-if)# exit
Switch(config)# ip route 0.0.0.0/0 10.10.11.1
Switch(config)# ip route 10.11.0.0 /16 10.11.11.1
Switch(config)# ipv6 route 0::/0 2001:DB8:2222:7270::1
```

```
Switch(config)# ipv6 route 2001:DB8:3333::/48  
2001:DB8:3333:7271::1
```

ネイバールータ 10.10.11.1 は、次のスタティック ルートを使用して設定する必要があります：  
ip ルート 172.25.13.2/32 10.10.10.2。

ネイバールータ 10.11.11.1 は、次のスタティック ルートを使用して設定する必要があります：  
ip ルート 172.25.13.2/32 10.11.11.2。

VLAN 1 に接続されたネイバー ルータ 2001:DB8:2222:7270::1 は、次のスタティック ルートを使用して設定する必要があります。

**ipv6 route 2001:DB8:2222:7272::72/128 2001:DB8:2222:7270::2312**

VLAN 1 に接続されたネイバー ルータ 2001:DB8:3333:7271::1 は、直下のスタティック ルートを使用して設定する必要があります。

IPv6 Route 2001:DB8:2222:7272::72/128 2001:DB8:3333:7271::2312

## CLIによるポートの管理

スタッキングをサポートするユニットのポートインターフェイスにアクセスするには、「interface GigabitEthernet X/0/Z (1gig インターフェイスの場合)」または「interface TenGigabitEthernet X/0/Y (10gig ポートの場合)」と入力します。X (1 ~ 4) はスタック ID、Y はアップリンクポート番号 (1 ~ 4)、Z はダウンリンクポート番号で、Z はポート数が 48 未満のユニットでも 1 ~ 48 になります。

### 光ファイバケーブルとトランシーバ

シスコブランドは SFP モジュール一式を提供します。シスコのスイッチは他のサードパーティをサポートしていますが、特定の SFP モジュールと組み合わせて使用する光ファイバケーブルのタイプに注意することが重要です。

光ファイバケーブルは、シングルモードとマルチモードの2つのタイプに分類できます。主な違いは、カバーできる距離と直径です。シングルモードファイバは、マルチモードよりも長い距離をカバーし、直径が小さい(約9マイクロメートル)です。マルチモードファイバの直径は 50 ~ 62.5 マイクロメートルです。

一方で、直接接続銅ケーブルの DAC ケーブルは、短距離に使用できます。カバーできる最大距離が 15 m であるため、主にマルチモード標準タイプのトランシーバを基準としています。ただし、AOC (アクティブ光ケーブル) は別です。

光ファイバ接続関連の問題をトラブルシューティングする場合は、SMF (シングルモードファイバケーブル) と MMF (マルチモードファイバケーブル)、および特定の光ファイバがサポートできる対応する SFP トランシーバを区別することが重要です。

シスコには、正しいペアリングを判断しようとする際に参照できるマトリックスがあります。次のリンクでは、Cisco 10gig SFP に関するインサイトを提供しています。

例：

Cisco SFP-10G-SR は直径 62.5 マイクロメートルの MMF タイプのケーブルでのみ動作し、Cisco SFP-10G-LR は SMF タイプのケーブルでのみ動作します。MMF の範囲は OM1 ~ OM5 です。OM とは光マルチモードのことです。OM1 タイプのケーブルの直径は 62.5 マイクロメートルですが、他のタイプ (OM2 ~ OM5) はすべて直径が 50 マイクロメートルです。

したがって、それらが混在しないようにするには、何が行われているかを知ることが重要です。

## リモート IP アドレスと OOB ポート

スイッチでは、OutOfBand (OOB) ポートで IP スタックがサポートされます。この IP スタックは ASIC ポートで実行している IP スタックとは切り離されており、特定のルートテーブルを設定する必要があります。

スイッチが複数の IP インターフェイスをサポートする場合、リモート IP アドレスまたは DNS 名を指定するときに、参照される IP スタックを指定する必要もあります。

## PHY 診断

次の例外が利用できます。

- 銅線ポート：PHY 診断は銅線ポートでのみサポートされます。
- 10 G ポート：動作ポートの速度が 10 G の場合、TDR テストがサポートされます。ケーブル長の分解能は 20 m です。

## CLI 出力修飾子

すべての **show** コマンドと **more** コマンド (**show technical support** を除く) では、出力修飾子が次のように追加されます。

```
<show/more command> | <output-modifier> <regular-expression-pattern>
```

出力修飾子は次のとおりです。

- **begin** : 指定した正規表現パターンに一致する文字列を含む最初の行から出力を開始します。
- **include** : 指定した正規表現パターンに一致する文字列を含む行のみを含めます。
- **exclude** : 指定した正規表現パターンに一致する文字列を含むすべての行を除外します。
- **count** : 指定した正規表現パターンに一致する文字列を含むすべての行をカウントし、結果を表示します (他の出力は表示されません)。



(注) 各コマンドで使用できる出力修飾子は1つのみです。入力したテキストの残りの部分は、正規表現パターンの一部になります。

正規表現は、パターン (フレーズ、番号、またはより複雑なパターン) です。CLI 文字列検索機能は、**show** コマンドまたは **more** コマンドの出力に正規表現を照合します。正規表現では、大文字と小文字が区別され、複雑な一致要件を指定することが可能です。

正規表現は、単一文字パターンか複数文字パターンです。つまり、正規表現は、コマンド出力中の同じ1文字に一致する1つの文字か、コマンド出力中の同じ複数の文字に一致する複数の文字です。コマンド出力中のパターンをストリングと呼びます。この項では、単一文字パターンと複数文字パターンの作成について説明します。また、量指定子、論理和指定子、位置指定子、カッコを使用した、より複雑な正規表現についても説明します。

### 単一文字パターン

最も単純な正規表現は、コマンド出力内の同じ1つの文字と一致する単一文字です。任意の文字 (A ~ Z, a ~ z) または数字 (0 ~ 9) を1文字のパターンとして使用できます。また、その他のキーボード文字 (「!」や「~」など) も1文字のパターンとして使用できますが、一部のキーボード文字は正規表現では特別な意味を持ちます。次の表に、特殊な意味を持つキーボード文字のリストを示します。

文字	意味
.	スペースを含む任意の単一文字と一致します。
*	0 個以上のパターンのシーケンスと一致します。
+	1 個以上のパターンのシーケンスと一致します。

文字	意味
?	0 または 1 回のパターンと一致します。
^	ストリングの先頭と一致します。
\$	ストリングの末尾と一致します。

これらの特殊文字を単一文字パターンとして使用するときは、各文字の前にバックスラッシュ (\) を置いて特別な意味を除外してください。

次の例は、それぞれドル記号、アンダースコア、プラス記号に一致する単一文字パターンマッチングの例です。

```
\$ \_ \+
```

単一文字パターンを範囲指定して、コマンド出力とのマッチングを行うことができます。たとえば、文字 `a`、`e`、`i`、`o`、`u` のいずれかを含むストリングに一致する正規表現を作成できます。パターンマッチングが成功するためには、これらの文字のいずれかだけがストリング中に存在する必要があります。1 文字のパターンの範囲を指定するには、1 文字のパターンを角カッコ ([ ]) で囲みます。たとえば、`[aeiou]` は小文字アルファベットの 5 つの母音のうちの任意の 1 文字と一致しますが、`[abcdABCD]` は小文字または大文字アルファベットの最初の 4 つの文字のうちの任意の 1 文字と一致します。

ダッシュ (-) で区切って範囲の終点だけを入力することにより範囲を簡略化することができます。

上の範囲は次のように単純化されます。

```
[a-dA-D]
```

ダッシュを範囲内の単一文字パターンとして追加するには、ダッシュをもう 1 つ追加し、その前にバックスラッシュを入力します。

```
[a-dA-D\]
```

次に示すように、右角カッコ (]) を、範囲内の単一文字パターンとして追加することもできます。

```
[a-dA-D\]]
```

上の例は、大文字または小文字のアルファベットの最初の 4 文字、ダッシュ、右角カッコのいずれかに一致します。範囲の先頭にキャレット (^) を追加することで、範囲の一致を反転させることができます。次の例は、その中の文字以外の文字に一致します。

```
[^a-dqsv]
```

次の例は、右角カッコ (]) または文字 `d` 以外のすべてと一致します。

```
[^\]d]
```



## 複数文字のパターン

正規表現を作成するとき、複数の文字を含むパターンを指定することもできます。複数文字正規表現は、文字、数字、特別な意味のないキーボード文字を組み合わせて作成します。たとえば、`a4%` は複数文字の正規表現です。

複数文字パターンでは、順序が大切です。`a4%` という正規表現は、`a` という文字のあとに `4` が続き、そのあとに `%` 記号が続く文字と一致します。ストリングの中に `a4%` という文字がその順序で含まれていないと、パターンマッチングは失敗します。複数文字の正規表現 `a.` ではピリオド文字に特別な意味があり、文字 `a` の後に続く 1 文字に相当します。この例では、`ab`、`a!`、または `a2` というストリングはすべてこの正規表現と一致します。

ピリオド文字の特別な意味を無効にするには、その前にバックスラッシュを挿入します。たとえば、表現 `a\.` がコマンド構文で使用されている場合、ストリング `a.` だけが一致します。

すべての文字、すべての数字、すべてのキーボード文字、文字と数字とその他のキーボード文字の組み合わせを含む複数文字正規表現を作成できます。たとえば、`telebit 3107 v32bis` は有効な正規表現です。

## 量指定子

指定した複数表現の出現を複数回一致させるようにシステムに指示する、より複雑な正規表現を作成できます。これを行うには、1 文字パターンと複数文字のパターンを使用していくつかの特殊文字を使用します。表 1 に、正規表現の出現回数を指定する特殊文字のリストを示します。

表 1: 表 1: 量指定子として使用する特殊文字

文字	説明
*	0 以上の単一文字パターンまたは複数文字パターンと一致します。
+	1 以上の単一文字パターンまたは複数文字パターンと一致します。
?	1 以上の単一文字パターンまたは複数文字パターンの 0 回または 1 回の出現と一致します。

次の例は、空文字を含む文字 `a` の任意の回数の出現と一致します。

`a*`

次のパターンでは、ストリングが一致するためには、文字 `a` が少なくとも 1 文字含まれていることが必要です。

`a+`

次のパターンは、ストリング `bb` または `bab` と一致します。

`ba?b`

次のストリングは、任意の数のアスタリスク (\*) と一致します。

\\*\*

乗算子を複数文字パターンと共に使用するには、パターンをカッコで囲みます。次の例で、パターンは複数文字ストリング `ab` の任意の回数の出現と一致します。

`(ab)*`

次のパターンは、英数字ペアの1つ以上のインスタンスに一致しますが、存在しない場合には一致しません（空の文字列とは一致しません）。

`([A-Za-z][0-9])+`

量指定子（\*、+、または?）を使用した一致の順序は、最長構造優先です。ネストした構造は、外側から内側に一致します。連結された構造は、構造の左側から一致します。したがって、上記の正規表現は `A9b3` と一致しますが、数字の前に文字が指定されているため `9Ab3` とは一致しません。

## 代替

選択を使用すると、ストリングに対して一致する代替パターンを指定できます。選択パターンは垂直線（|）で区切ります。選択肢のいずれか1つだけがストリングと一致します。たとえば、正規表現 `codex|telebit` は文字列 `codex` または文字列 `telebit` のいずれかに一致しますが、`codex` と `telebit` の両方には一致しません。

## 位置指定

正規表現パターンを文字列の先頭または末尾と一致させるようにシステムに指示することができます。文字列の一部にこれらの正規表現を位置指定するには、表2に示す特殊文字を使用します。

表 2: 表 2: 位置指定子として使用する特殊文字

文字	説明
<code>^</code>	ストリングの先頭と一致します。
<code>\$</code>	ストリングの末尾と一致します。

たとえば、正規表現 `^con` は `con` で始まるストリングに一致し、`$sole` は `sole` で終わるストリングに一致します。

文字列の先頭を示すのに加えて、`^` 記号は角カッコの中で使用された場合は論理関数 `not` を示すものとして使用できます。たとえば、正規表現 `[^abcd]` は、`a`、`b`、`c`、または `d` 以外の任意の単一文字に一致する範囲を示します。

## 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。