



802.1Q VLAN インターフェイスの設定

VLAN とは、実際は異なる LAN セグメント上のデバイスでも、同じセグメントで接続している場合と同様に通信できるように設定された、1 つまたは複数の LAN 上にあるデバイスのグループです。VLAN は、物理接続ではなく論理グループに基づいているため、ユーザ、ホスト管理、帯域幅割り当て、リソースの最適化に非常に柔軟に対応します。

IEEE 802.1Q プロトコル規格では、ブロードキャストおよびマルチキャストのトラフィックが必要以上の帯域を消費しないように、大規模なネットワークを小規模なパーツに分割することで問題に対処しています。また、内部ネットワークのセグメント間に、より高レベルのセキュリティを実現できます。

802.1Q 仕様は、イーサネット フレームに VLAN メンバーシップ情報を挿入する標準方式を確立します。Cisco NCS 5000 シリーズルータは、10 ギガビットイーサネット インターフェイスおよび 100 ギガビットイーサネット インターフェイス上で VLAN のサブインターフェイスの設定をサポートします。VLAN の範囲は 1 ~ 4094 です。

802.1Q タグ付きフレーム

IEEE 802.1Q タグ ベースの VLAN は、MAC ヘッダーの特別なタグを使用し、ブリッジでのフレームの VLAN メンバーシップを識別できます。このタグは、VLAN および Quality of Service (QoS) のプライオリティの識別に使用されます。VLAN ID は、フレームを特定の VLAN に関連付けて、スイッチがネットワークでフレームを処理する必要があるという情報を提供します。タグ付きフレームは、タグなしフレームよりも 4 バイト長く、イーサネット フレームの Type および Length フィールドにある 2 バイトの Tag Protocol Identifier (TPID) フィールドと、イーサネット フレームの Source Address フィールドの後ろから始まる 2 バイトの Tag Control Information (TCI) が含まれます。

802.1Q タグ付きフレームの詳細については、『*L2VPN and Ethernet Services Configuration Guide for Cisco NCS 5000 Series Routers*』の「*References for Carrier Ethernet Model*」の項を参照してください。

- [802.1Q VLAN インターフェイスの設定方法 \(2 ページ\)](#)
- [802.1Q VLAN インターフェイスの設定に関する情報 \(7 ページ\)](#)

802.1Q VLAN インターフェイスの設定方法

ここでは、次の手順について説明します。

802.1Q VLAN サブインターフェイスの設定

ここでは、802.1Q VLAN サブインターフェイスの設定手順について説明します。これらのサブインターフェイスを削除するには、「802.1Q VLAN サブインターフェイスの削除」の項を参照してください。

手順

ステップ1 **configure**

例：

```
RP/0/RP0/cpu 0: router# configure
```

グローバル コンフィギュレーション モードを開始します。

ステップ2 **interface {TenGigE | FortyGigE | HundredGigE | Bundle-Ether} interface-path-id.subinterface**

例：

```
RP/0/RP0/cpu 0: router(config)# interface TenGigE 0/0/0/4.10
```

サブインターフェイス コンフィギュレーション モードを開始し、インターフェイス タイプ、ロケーション、サブインターフェイス番号を指定します。

- *interface-path-id* 引数を、次のいずれかのインスタンスに置き換えます。
- 物理イーサネット インターフェイス インスタンスまたはイーサネット バンドル インスタンス。名前表記は *rack/slot/module/port* で、値の間のスラッシュは表記の一部として必要です。
- イーサネット バンドル インスタンス。範囲は 1 ~ 65535 です。
- *subinterface* 引数をサブインターフェイスの値に置き換えます。範囲は 0 ~ 2147483647 です。
- 名前表記は *interface-path-id.subinterface* で、表記の一部として引数をピリオドで区切る必要があります。

ステップ3 **encapsulation dot1q**

例：

```
RP/0/RP0/cpu 0: router(config-subif)# encapsulation dot1q 100
```

インターフェイスのレイヤ 2 カプセル化を設定します。

ステップ 4 `ipv4 address ip-address mask`

例：

```
RP/0/RP0/cpu 0: router(config-subif)# ipv4 address 178.18.169.23/24
```

IP アドレスおよびサブネット マスクをサブインターフェイスに割り当てます。

- `ip-address` をインターフェイスのプライマリ IPv4 アドレスに置き換えます。
- `mask` を関連付けられた IP サブネットのマスクに置き換えます。ネットワーク マスクは、次のいずれかの方法で指定できます。
- 4 分割ドット付き 10 進表記のアドレスでネットワーク マスクを指定します。たとえば、255.0.0.0 は、値が 1 の各ビットは、対応するアドレスのビットがそのネットワークアドレスに属することを示します。
- ネットワークマスクは、スラッシュ (/) と数字で示すことができます。たとえば、/8 は、マスクの最初の 8 ビットが 1 で、対応するアドレスのビットがネットワークアドレスであることを示します。

ステップ 5 `exit`

例：

```
RP/0/RP0/cpu 0: router(config-subif)# exit
```

(任意) サブインターフェイス コンフィギュレーション モードを終了します。

- `exit` コマンドは、明示的に指定する必要はありません。

ステップ 6 ステップ 2～5 を繰り返し、残りの VLAN サブインターフェイスを定義します。

—

ステップ 7 `end` または `commit`

例：

```
RP/0/RP0/cpu 0: router(config)# end
```

または

```
RP/0/RP0/cpu 0: router(config)# commit
```

設定変更を保存します。

- `end` コマンドを実行すると、次に示す変更のコミットを求めるプロンプトが表示されません。

```
Uncommitted changes found, commit them before exiting(yes/no/cancel)?  
[cancel]:
```

- **yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。

-no と入力すると、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。

-cancel と入力すると、ルータは現在のコンフィギュレーションセッションで継続されます。コンフィギュレーションセッションは終了せず、設定変更もコミットされません。

- 実行コンフィギュレーションファイルに設定変更を保存し、コンフィギュレーションセッションを継続するには、**commit** コマンドを使用します。

ステップ 8 show ethernet trunk bundle-ether instance

例：

```
RP/0/RP0/cpu 0: router# show ethernet trunk bundle-ether 5
```

(任意) インターフェイス コンフィギュレーションを表示します。

イーサネットバンドルインスタンスの範囲は 1 ~ 65535 です。

確認

次に、イーサネット インターフェイスの設定を確認する例を示します。

```
# show ethernet trunk be 1020 Wed May 17 16:43:32.804 EDT
```

Trunk Interface	St Ly	MTU	Subs	Sub types		Sub states		
				L2	L3	Up	Down	Ad-Down
BE1020	Up L3	9100	3	3	0	3	0	0
Summary			3	3	0	3	0	0

VLAN での接続回線の設定

VLAN で接続回線を設定するには、次の手順で操作します。

手順

ステップ 1 configure

例：

```
RP/0//CPU0:router# configure
```

グローバル コンフィギュレーション モードを開始します。

ステップ 2 interface [GigabitEthernet | TenGigE | Bundle-Ether | FortyGigE] interface-path] id.subinterface l2transport

例：

```
RP/0//CPU0:router(config)# interface TenGigE 0/0/0/1.1 l2transport
```

サブインターフェイス コンフィギュレーション モードを開始し、インターフェイス タイプ、ロケーション、サブインターフェイス番号を指定します。

- *interface-path-id* 引数を、次のいずれかのインスタンスに置き換えます。
- 物理イーサネット インターフェイス インスタンスまたはイーサネットバンドル インスタンス。名前表記は *rack/slot/module/port* で、値の間のスラッシュは表記の一部として必要です。
- イーサネットバンドル インスタンス。範囲は 1 ~ 65535 です。
- *subinterface* 引数をサブインターフェイスの値に置き換えます。範囲は 0 ~ 4095 です。
- 名前の表記は *instance.subinterface* の形式で、表記の一部として引数をピリオドで区切る必要があります。
- コマンド文字列に **l2transport** キーワードを含める必要があります。そうしないと、ACではなく、レイヤ 3 サブインターフェイスが作成されます。

ステップ 3 encapsulation dot1q 100

例：

```
RP/0//CPU0:router (config-subif)# encapsulation dot1q 100
```

インターフェイスのレイヤ 2 カプセル化を設定します。

(注) **dot1q vlan** コマンドは、**encapsulation dot1q** コマンドに置き換えられます。引き続き、下位互換性のために使用可能ですが、レイヤ 3 インターフェイスだけが対象です。

ステップ 4 end または commit

例：

```
RP/0//CPU0:router(config-if-12)# end
```

または

```
RP/0//CPU0:router(config-if-12)# commit
```

設定変更を保存します。

- **end** コマンドを実行すると、次に示す変更のコミットを求めるプロンプトが表示されません。

```
Uncommitted changes found, commit them before exiting(yes/no/cancel)?  
[cancel]:
```

-**yes** と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。

-**no** と入力すると、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。

- **cancel** と入力すると、ルータは現在のコンフィギュレーションセッションで継続されず、コンフィギュレーションセッションは終了せず、設定変更もコミットされません。
- 実行コンフィギュレーションファイルに設定変更を保存し、コンフィギュレーションセッションを継続するには、**commit** コマンドを使用します。

ステップ 5 `show interfaces [GigabitEthernet|FortyGigE|Bundle-Ether|TenGigE] interface-path-id.subinterface`

例：

```
RP/0//CPU0:router# show interfaces TenGigE 0/0/0/3.1
```

(任意) ルータ上のインターフェイスに関する統計情報を表示します。

802.1Q VLAN サブインターフェイスの削除

ここでは、このモジュールの「802.1Q VLAN サブインターフェイスの設定」の項で設定した 802.1Q VLAN サブインターフェイスを削除する方法について説明します。

手順

ステップ 1 `configure`

例：

```
RP/0/RP0/cpu 0: router# configure
```

グローバル コンフィギュレーション モードを開始します。

ステップ 2 `no interface {TenGigE|FortyGigE|HundredGigE|Bundle-Ether} interface-path-id.subinterface`

例：

```
RP/0/RP0/cpu 0: router(config)# no interface TenGigE 0/0/0/4.10
```

サブインターフェイスを削除すると、そのサブインターフェイスに適用されているすべての設定も自動的に削除されます。

- *instance* 引数を次のインスタンスのいずれかで置き換えます。
- 物理イーサネット インターフェイス インスタンスまたはイーサネット バンドル インスタンス。名前表記は *rack/slot/module/port* で、値の間のスラッシュは表記の一部として必要です。
- イーサネット バンドル インスタンス。範囲は 1 ~ 65535 です。
- *subinterface* 引数をサブインターフェイスの値に置き換えます。範囲は 0 ~ 2147483647 です。

名前の表記は *instance.subinterface* の形式で、表記の一部として引数をピリオドで区切る必要があります。

ステップ 3 ステップ 2 を繰り返し、その他の VLAN サブインターフェイスを削除します。

—

ステップ 4 **end** または **commit**

例：

```
RP/0/RP0/cpu 0: router(config)# end
```

または

```
RP/0/RP0/cpu 0: router(config)# commit
```

設定変更を保存します。

- **end** コマンドを実行すると、次に示す変更のコミットを求めるプロンプトが表示されません。

```
Uncommitted changes found, commit them before exiting(yes/no/cancel)?  
[cancel]:
```

- **yes** と入力すると、実行コンフィギュレーションファイルに変更が保存され、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。

- **no** と入力すると、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。

- **cancel** と入力すると、ルータは現在のコンフィギュレーションセッションで継続されます。コンフィギュレーションセッションは終了せず、設定変更もコミットされません。

- 実行コンフィギュレーションファイルに設定変更を保存し、コンフィギュレーションセッションを継続するには、**commit** コマンドを使用します。

802.1Q VLAN インターフェイスの設定に関する情報

802.1Q VLAN インターフェイスを設定するには、次の概念を理解する必要があります。

サブインターフェイス

サブインターフェイスは、ハードウェアインターフェイス上に作成される論理インターフェイスです。これらのソフトウェア定義のインターフェイスにより、単一のハードウェアインターフェイス上でトラフィックを論理チャンネルに分割することができ、また、物理またはバンドルインターフェイス上で帯域幅を効率的に利用することができます。

サブインターフェイスは、インターフェイス名の末尾に拡張を追加することで、他のインターフェイスと区別されます。たとえば、物理インターフェイス TenGigE 0/0/0/0 上のイーサネットサブインターフェイス 23 は、TenGigE 0/0/0/0.23 となります。

サブインターフェイスがトラフィックを渡すことができるようにするには、有効なタグ付きプロトコルのカプセル化と VLAN 識別子の割り当てが必要です。すべてのイーサネットサブインターフェイスは常に、デフォルトで 802.1Q VLAN でカプセル化されます。ただし、VLAN 識別子は明示的に定義する必要があります。

サブインターフェイスに適用可能なスケール値は次のとおりです。

- システムあたりのサブインターフェイス = 1024
- ラインカードあたりのサブインターフェイス = 1024
- NPU あたりのサブインターフェイス = 1024
- インターフェイスあたりのサブインターフェイス = 512
- コアあたりのサブインターフェイス = 512

サブインターフェイス MTU

サブインターフェイスの最大伝送単位 (MTU) は、物理インターフェイスから継承されます。これには、802.1Q VLAN タグに許可されている追加の 4 バイトも含まれます。MTU が設定されていない場合、デフォルトのサブインターフェイスは物理インターフェイスの MTU を継承します。サブインターフェイスには NPU あたり最大 3 つの異なる MTU を使用できます。

EFP

イーサネットフローポイント (EFP) は、抽象的なルータのアーキテクチャを説明する Metro Ethernet Forum (MEF) の用語です。EFP は VLAN カプセル化を使用した レイヤ 2 サブインターフェイスによって実装されます。用語 EFP は VLAN タグ付き L2 サブインターフェイスと同義的に使用されます。

VLAN でのレイヤ 2 VPN

レイヤ 2 バーチャルプライベート ネットワーク (L2VPN) 機能を利用すると、サービスプロバイダー (SP) は、地理的に離れたカスタマー サイトにレイヤ 2 サービスを提供できるようになります。

VLAN 接続回線 (AC) を設定するための設定モデルは、基本の VLAN の設定に使用するモデルに類似しています。ユーザはまず VLAN サブインターフェイスを作成し、次にサブインターフェイスコンフィギュレーションモードで VLAN を設定します。AC を作成するには、**interface** コマンド文字列に **l2transport** キーワードを含めて、そのインターフェイスがレイヤ 2 インターフェイスであることを指定する必要があります。

VLAN AC は、これらの L2VPN 操作のモードをサポートします。

- 基本の Dot1Q AC : AC は、特定の VLAN タグで送受信されるすべてのフレームに対応します。
- QinQ AC : AC は、特定の外部 VLAN タグおよび特定の内部 VLAN タグで送受信されるすべてのフレームに対応します。QinQ は、2 つのタグのスタックを使用する Dot1Q の拡張です。

CE-to-PE リンクの各 VLAN は、（VC タイプ 4 または VC タイプ 5 を使用する）独立した L2VPN 接続として設定できます。

