Cisco ASR 9000 シリーズ ルータでの 802.1Q VLAN インターフェイスの設定

このモジュールでは、Cisco ASR 9000 シリーズ アグリゲーション サービス ルータの 802.1Q VLAN インターフェイスの設定と管理について説明します。

IEEE 802.1Q 仕様は、VLAN メンバーシップ情報のあるタグ付きイーサネット フレームの標準方式を確立し、ブリッジド LAN インフラストラクチャ内にある VLAN トポロジーの定義、操作、および管理ができる VLAN ブリッジの動作を定義します。

802.1Q 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することを目的としています。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 VLAN インターフェイス設定の機能履歴

リリース	変更内容
リリース 3.7.2	この機能は、Cisco ASR 9000 シリーズ ルータで導入されました。
リリース 3.9.0	レイヤ 2 dot1q が更新されました。カプセル化 dot1q が追加されました。

内容

- 「802.1Q VLAN インターフェイス設定の前提条件」（P.685）
- 「802.1Q VLAN インターフェイスの設定に関する情報」（P.686）
- 「802.1Q VLAN インターフェイスの設定方法」（P.689）
- 「VLAN インターフェイスの設定例」（P.695）
- 「その他の関連資料」（P.697）

802.1Q VLAN インターフェイス設定の前提条件

適切なタスク ID を含むタスク グループに関連付けられているユーザ グループに属している必要があります。このコマンド リファレンスには、各コマンドに必要なタスク ID が含まれます。ユーザ グループの割り当てが原因でコマンドを使用できないと考えられる場合、AAA 管理者に連絡してください。

802.1Q VLAN インターフェイスを設定する前に、次の条件を満たしていることを確認してください。

- ギガビット イーサネット インターフェイス、10 ギガビット イーサネット インターフェイス、またはイーサネット バンドル インターフェイスの設定が完了している必要があります。

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、次の概念を理解しておく必要があります。

- 「802.1Q VLAN の概要」（P.686）
- 「802.1Q タグ付きフレーム」（P.686）
- 「802.1Q VLAN インターフェイスの CFM」（P.686）
- 「サブインターフェイス」（P.687）
- 「サブインターフェイス MTU」（P.687）
- 「ネイティブ VLAN」（P.687）
- 「EFP」（P.687）
- 「VLAN インターフェイスでのレイヤ 2 VPN」（P.687）
- 「他のレイヤ 2 VPN 機能」（P.688）

802.1Q VLAN の概要

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接続に基づいているため、ユーザ管理、ホスト管理、帯域割り当て、およびリソースの最適化がとても柔軟です。

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 IOS XR ソフトウェアは、ギガビット イーサネットおよび 10 ギガビット イーサネット インターフェイスでの VLAN サブインターフェイスの設定をサポートしています。

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 は、手動での入力によってスタティックに作成することも、Generic Attribute Registration Protocol（GARP）VLAN Registration プロトコル（GVRP）を介してダイナミックに作成することもできます。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 VLAN インターフェイスの CFM

802.1Q VLAN インターフェイスをモニタするための接続障害管理（CFM）の設定方法は、イーサネット インターフェイスをモニタするための CFM の設定と同じです。

イーサネット インターフェイスの CFM の設定については、「Cisco ASR 9000 シリーズ ルータのイーサネット OAM の設定」モジュールの次のセクションを参照してください。

- 「イーサネット CFM」（P.68）
- 「イーサネット CFM の設定」（P.109）
- 「イーサネット CFM サービスの設定：例」（P.167）
- 「イーサネット CFM の show コマンド：例」（P.168）

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ネルに分割することができ、また、物理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 0/1/0/0 上のイーサネット サブインターフェイス 23 は、TenGigE 0/1/0/0.23 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 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。

ネイティブ VLAN

Cisco ASR 9000 シリーズ ルータは、ネイティブ VLAN をサポートしません。ただし、同等の機能は、次のように **encapsulation** コマンドを使用して達成されます。

```
encapsulation dot1q TAG-ID, untagged
```

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。Cisco ASR 9000 シリーズ ルータでは、EFP は VLAN カプセル化を使用した L2 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 インターフェイスでのレイヤ 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、L2VPN 操作の 3 つ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。
- Q-in-Any AC：AC は、内部 VLAN タグが L3 終端でない限り、特定の外部 VLAN タグおよび任意の内部 VLAN タグで送受信されるすべてのフレームに対応します。Q-in-Any は、ワイルドカード化を使用して任意の 2 番目のタグに一致させる QinQ の拡張です。

（注） Q-in-Any モードは、基本の Dot1Q モードを変化させたものです。Q-in-Any モードでは、フレームは基本の QinQ カプセル化が行われていますが、Q-in-Any モードでは内部タグは意味を持ちません。ただし、いくつかの特定の内部 VLAN タグは、特定のサービスに使用されます。たとえば、一般的なインターネット アクセスに L3 サービスを提供するために、あるタグが使用されることがあります。

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。VLAN に L2VPN を設定するには、「VLAN での接続回線の設定」（P.691）を参照してください。

VLAN に L2VPN を設定する場合は、次の事項に注意する必要があります。

- Cisco IOS XR ソフトウェアは LC ごとに 4k AC をサポートします。
- ポイントツーポイント接続では、2 つの AC を同じタイプにするべきではありません。たとえば、ポート モードのイーサネット AC を、Dot1Q イーサネット AC に接続することができます。
- 疑似回線は、VLAN モードまたはポート モードで実行できます。VLAN モードで実行される疑似回線に単一の Dot1Q タグを設定することができますが、ポート モードで実行される疑似回線にタグを設定することはできません。これらの異なるタイプの回路を接続するには、インターワーキングが必要です。この場合のインターワーキングは、タグのポップ、プッシュ、書き換えの形を取ります。レイヤ 2 VPN を使用するメリットは、まったく異なるタイプのメディアを接続するのに必要なインターワーキングを簡素化できることにあります。
- MPLS 疑似回線の両側にある AC は、異なるタイプでもかまいません。この場合は、AC の一方または両方のエンドで、疑似回線接続を行うための適切な変換が行われます。

AC と疑似回線の情報を表示するには、**show interfaces** コマンドを使用します。

（注） L2VPN ネットワークの設定の詳細については、『*Cisco ASR 9000 Series Router Multiprotocol Label Switching Configuration Guide*』の「*Implementing MPLS Layer 2* VPNs」モジュールを参照してください。

他のレイヤ 2 VPN 機能

次のレイヤ 2 VPN 機能の詳細については、『*Cisco ASR 9000 Series Aggregation Services Router L2VPN and Ethernet Services Configuration Guide*』および『*Cisco ASR 9000 Series Aggregation Services Router L2VPN and Ethernet Services Command Reference*』を参照してください。

- プロバイダー バックボーン ブリッジ（PBB）802.1ah
- Policy-Based Forwarding（PBF）
- MVRP 802.1（MVRP-lite）

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」（P.689）
- 「VLAN での接続回線の設定」（P.691）
- 「802.1Q VLAN サブインターフェイスの削除」（P.694）

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、このモジュールの「802.1Q VLAN サブインターフェイスの削除」を参照してください。

手順の概要

1. **configure**
2. **interface** {**GigabitEthernet** | **TenGigE** |**Bundle-Ether**} *interface-path-id.subinterface*
3. **encapsulation dot1q**
4. **ipv4 address** *ip-address mask*
5. **exit**
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。
7. **end**
   または
   commit
8. **show ethernet trunk bundle-ether** *instance*（任意）

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/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**} *interface-path-id.subinterface* **例：** `RP/0/RSP0/CPU0:router(config)# interface TenGigE 0/2/0/4.10`	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。名前の表記は *rack*/*slot*/*module*/*port* の形式で、表記の一部として値をスラッシュで区切る必要があります。 – 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値に置き換えます。範囲は 0 ～ 4095 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/RSP0/CPU0:router(config-subif)# encapsulation dot1q 100, untagged`	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、Cisco ASR 9000 シリーズ ルータでは **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	**ipv4 address** *ip-address mask* **例：** `RP/0/RSP0/CPU0: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。 – スラッシュ（/）と数字による表記。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。

	コマンドまたはアクション	目的
ステップ 5	**exit** **例：** `RP/0/RSP0/CPU0: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/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		設定変更を保存します。
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		`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 ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、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、現在のコンフィギュレーション セッションが継続します。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/RSP0/CPU0: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。

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。

手順の概要

1. **configure**
2. **interface** {**GigabitEthernet** | **TenGigE** | **Bundle-Ether**] *interface-path-id.subinterface* **l2transport**
3. **encapsulation dot1q**
4. **l2protocol cpsv** {**tunnel** | **reverse-tunnel**}

5. **end**
   または
   commit
6. **show interfaces** [**GigabitEthernet** | **TenGigE**]

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/CPU0:router# configure terminal`	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** | **TenGigE**] *interface-path*] *id.subinterface* **l2transport** **例：** `RP/0/RSP0/CPU0:router(config)# interface TenGigE 0/1/0/0.1 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。 • 引数を、次のいずれかのインスタンスで置き換えます。 – 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。名前の表記は *rack*/*slot*/*module*/*port* の形式で、表記の一部として値をスラッシュで区切る必要があります。 – 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。 **（注）** 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** **例：** `RP/0/RSP0/CPU0:router(config-subif)# encapsulation dot1q 100, untagged`	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、Cisco ASR 9000 シリーズ ルータでは **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。

	コマンドまたはアクション	目的
ステップ 4	**l2protocol cpsv** {**tunnel** | **reverse-tunnel**} **例：** `RP/0/RSP0/CPU0:router(config-if-l2)# l2protocol cpsv tunnel`	プロトコル CDP、PVST+、STP、VTP のイーサネット インターフェイスでのレイヤ 2 プロトコル トンネリングとプロトコル データ ユニット（PDU）フィルタリングを設定します。 • **tunnel**：インターフェイスに入るときのフレームの L2PT カプセル化と、インターフェイスから出るときのフレームのカプセル化解除を指定します。 • **reverse-tunnel**：インターフェイスから出るときのフレームの L2PT カプセル化と、インターフェイスに入るときのフレームのカプセル化解除を指定します。

ステップ 5	**end**
	または
	commit
	
	例：
	`RP/0/RSP0/CPU0:router(config-if-l2)# end`
	または
	`RP/0/RSP0/CPU0:router(config-if-l2)# commit`
		設定変更を保存します。
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		`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 ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、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、現在のコンフィギュレーション セッションが継続します。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 6	**show interfaces** [**GigabitEthernet** | **TenGigE**] *interface-path-id.subinterface* **例：** `RP/0/RSP0/CPU0:router# show interfaces TenGigE 0/3/0/0.1`	（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。

次の作業

- ポイントツーポイント疑似回線相互接続を AC 上で設定するには、『*Cisco ASR 9000 Series Router Multiprotocol Label Switching Configuration Guide*』の「*Implementing MPLS Layer 2* VPNs」モジュールを参照してください。
- マルチプロトコル ラベル スイッチング（MPLS）や QoS などのレイヤ 3 サービス ポリシーを VLAN に付加する方法については、該当する Cisco ASR 9000 シリーズ ルータのコンフィギュレーション ガイドを参照してください。

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タスクで設定した 802.1Q VLAN サブインターフェイスを削除する手順について説明します。

手順の概要

1. **configure**
2. **no interface** {**GigabitEthernet** | **TenGigE** | **Bundle-Ether**] *interface-path-id.subinterface*
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。
4. **end**
   または
   commit
5. **show ethernet trunk bundle-ether** *instance*（任意）

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**configure** **例：** `RP/0/RSP0/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	**no interface** {**GigabitEthernet** | **TenGigE** | **Bundle-Ether**] *interface-path-id.subinterface* **例：** `RP/0/RSP0/CPU0:router(config)# no interface TenGigE 0/2/0/4.10`	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、次のいずれかのインスタンスで置き換えます。 – 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。名前の表記は *rack*/*slot*/*module*/*port* の形式で、表記の一部として値をスラッシュで区切る必要があります。 – 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。

	コマンドまたはアクション	目的
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/RSP0/CPU0:router(config)# end`
	または
	`RP/0/RSP0/CPU0:router(config)# commit`
		設定変更を保存します。
		• **end** コマンドを実行すると、変更をコミットするように要求されます。
		`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 ファイルに設定変更が保存され、コンフィギュレーション セッションが終了し、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、現在のコンフィギュレーション セッションが継続します。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 ethernet trunk bundle-ether** *instance* **例：** `RP/0/RSP0/CPU0: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。

VLAN インターフェイスの設定例

ここでは、次の例について説明します。

「VLAN サブインターフェイス：例」（P.695）

VLAN サブインターフェイス：例

次の例では、3 つの VLAN サブインターフェイスを一度に作成する方法を示します。

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface TenGigE 0/2/0/4.1
RP/0/RSP0/CPU0:router(config-subif)# encapsulation dot1q 100
RP/0/RSP0/CPU0:router(config-subif)# ipv4 address 10.0.10.1/24
RP/0/RSP0/CPU0:router(config-subif)# interface TenGigE0/2/0/4.2
RP/0/RSP0/CPU0:router(config-subif)# encapsulation dot1q 101
RP/0/RSP0/CPU0:router(config-subif)# ipv4 address 10.0.20.1/24
RP/0/RSP0/CPU0:router(config-subif)# interface TenGigE0/2/0/4.3
RP/0/RSP0/CPU0:router(config-subif)# encapsulation dot1q 102
```

```
RP/0/RSP0/CPU0:router(config-subif)# ipv4 address 10.0.30.1/24
RP/0/RSP0/CPU0:router(config-subif)# commit
RP/0/RSP0/CPU0:router(config-subif)# exit
RP/0/RSP0/CPU0:router(config)# exit

RP/0/RSP0/CPU0:router# show ethernet trunk bundle-Ether 1
Trunk                                Sub types        Sub states
Interface     St Ly    MTU    Subs       L2      L3       Up    Down  Ad-Down
BE1           Up L3   1514    1000        0    1000     1000       0       0

Summary                       1000        0    1000     1000       0       0
```

次の例では、1 つのイーサネット バンドルに対して 2 個の VLAN サブインターフェイスを作成する方法を示します。

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface bundle-ether 2
RP/0/RSP0/CPU0:router(config-if)# ipv4 address 192.168.2.1/24
RP/0/RSP0/CPU0:router(config-if)# exit
RP/0/RSP0/CPU0:router(config)# interface bundle-ether 2.1
RP/0/RSP0/CPU0:router(config-subif)# encapsulation dot1q 100
RP/0/RSP0/CPU0:router(config-subif)# ipv4 address 192.168.100.1/24
RP/0/RSP0/CPU0:router(config-subif)# exit
RP/0/RSP0/CPU0:router(config)# interface bundle-ether 2.2
RP/0/RSP0/CPU0:router(config-subif)# encapsulation dot1q 200
RP/0/RSP0/CPU0:router(config-subif)# ipv4 address 192.168.200.1/24
RP/0/RSP0/CPU0:router(config-subif)# exit
RP/0/RSP0/CPU0:router(config)# commit
```

次に、基本的な dot1Q AC を作成する例を示します。

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet 0/0/0/0.1
RP/0/RSP0/CPU0:router(config-subif)# l2transport
RP/0/RSP0/CPU0:router(config-subif)# encapsulation dot1q 100
RP/0/RSP0/CPU0:router(config-subif)# commit
RP/0/RSP0/CPU0:router(config-subif)# exit
RP/0/RSP0/CPU0:router(config)# exit
```

次の例では、Q-in-Q AC を作成する方法を示します。

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet 0/0/0/0.2
RP/0/RSP0/CPU0:router(config-subif)# l2transport
RP/0/RSP0/CPU0:router(config-subif)# encapsulation dot1q 200 second-dot1q 201
RP/0/RSP0/CPU0:router(config-subif)# commit
RP/0/RSP0/CPU0:router(config-subif)# exit
RP/0/RSP0/CPU0:router(config)# exit
```

次の例では、Q-in-Any AC を作成する方法を示します。

```
RP/0/RSP0/CPU0:router# configure
RP/0/RSP0/CPU0:router(config)# interface GigabitEthernet 0/0/0/0.3
RP/0/RSP0/CPU0:router(config-subif)# l2transport
RP/0/RSP0/CPU0:router(config-subif)# encapsulation dot1q 300 second-dot1q any
RP/0/RSP0/CPU0:router(config-subif)# commit
RP/0/RSP0/CPU0:router(config-subif)# exit
RP/0/RSP0/CPU0:router(config)# exit
```

その他の関連資料

ここでは、VLAN インターフェイスの設定に関連する参考資料を示します。

関連資料

関連項目	参照先
Cisco ASR 9000 シリーズ ルータ マスター コマンド リファレンス	*『Cisco ASR 9000 Series Router Master Commands List』*
Cisco ASR 9000 シリーズ ルータ インターフェイス コンフィギュレーション コマンド	*『Cisco ASR 9000 Series Router Interface and Hardware Component Command Reference』*
Cisco IOS XR ソフトウェアを使用する Cisco ASR 9000 シリーズ ルータの初期システム ブートアップと設定に関する情報。	*『Cisco ASR 9000 Series Router Getting Started Guide』*
ユーザ グループとタスク ID に関する情報	*『Cisco ASR 9000 Series Router Interface and Hardware Component Command Reference』*

標準

標準	タイトル
この機能によってサポートされる新しい規格または変更された規格はありません。またこの機能による既存規格のサポートに変更はありません。	—

MIB

MIB	MIB のリンク
このモジュールに適用できる MIB はありません。	Cisco IOS XR ソフトウェアを使用して選択したプラットフォームの MIB を検索およびダウンロードするには、次の URL の Cisco MIB Locator を使用します。 http://cisco.com/public/sw-center/netmgmt/cmtk/mibs.shtml

RFC

RFC	タイトル
この機能によりサポートされた新規 RFC または改訂 RFC はありません。またこの機能による既存 RFC のサポートに変更はありません。	—

シスコのテクニカル サポート

説明	リンク
シスコのテクニカル サポート Web サイトでは、製品、テクノロジー、ソリューション、技術的なヒント、およびツールへのリンクなどの、数千ページに及ぶ技術情報が検索可能です。Cisco.com に登録済みのユーザは、このページから詳細情報にアクセスできます。	http://www.cisco.com/en/US/support/index.html