IPSec 仮想トンネル インターフェイス

IPsec 仮想トンネル インターフェイス（VTI）では、IPsec トンネルを終了するためのルーティング可能なインターフェイス タイプと、オーバーレイ ネットワークを形成するためにサイト間の保護を定義する簡単な手段が提供されます。IPsec VTI によって、リモート リンクを保護するための IPsec の設定が簡素化され、マルチキャストがサポートされ、さらには、ネットワーク管理およびロード バランシングが簡単に実現できるようになります。

（注） セキュリティに対する脅威は、脅威からの保護に役立つ暗号化技術と同様に絶え間なく変化しています。シスコの暗号化に関する最新の推奨事項の詳細は、『Next Generation Encryption』（NGE）ホワイトペーパーを参照してください。

IPsec 仮想トンネル インターフェイスの制約事項

フラグメンテーション

フラグメンテーションは、IPsec トンネルではサポートされていません。ホストの MTU を小さく設定してパケットフラグメントを回避することや、任意のデバイスでパケットをフラグメント化することを選択できます。

IPsec トランスフォーム セット

IPsec トランスフォーム セットを設定できるのは、トンネル モードでだけです。

IKE セキュリティ アソシエーション

インターネット キー交換（IKE）セキュリティ アソシエーション（SA）は VTI にバインドされています。

IPsec SA トラフィック セレクタ

スタティック VTI では、VTI インターフェイスに接続している単一の IPsec SA だけがサポートされます。IPsec SA のトラフィック セレクタは常に "IP any any" です。

デフォルトでは、スタティック VTI（SVTI）は、仮想トンネルインターフェイスに接続された 1 つの IPsec SA のみをサポートします。IPsec SA のトラフィックセレクタは常に "IP any any" です。

IPv4 パケット

この機能は、IPv4 パケットをカプセル化するように設定された SVTI をサポートしますが、IPv4 パケットで IPv6 パケットを伝送したり、IPv6 パケットで IPv4 パケットを伝送したりすることはできません。

tunnel protection

IPsec IPv4 モードで **tunnel mode ipsec ipv4** コマンドを使用する場合は、**shared** キーワードを設定しないでください。

traceroute

VTI での暗号化オフロードを使用したトレースルート機能はサポートされていません。

VxLAN GPE トンネルインターフェイス

VxLAN GPE トンネルインターフェイスは、IPsec VTI と同じ送信元インターフェイスを使用できません。

IPsec 仮想トンネル インターフェイスに関する情報

IPsec VTI の使用により、リモートアクセスの保護を提供する必要がある場合の設定プロセスが簡素化され、カプセル化に Generic Routing Encapsulation（GRE）またはレイヤ 2 トンネリングプロトコル（L2TP）トンネルを使用する代替手段が提供されます。IPsec VTI を使用するメリットは、設定において物理インターフェイスに対する IPsec セッションのスタティックマッピングが必要ないことです。IPsec トンネル エンドポイントは実際（仮想）のインターフェイスに関連付けられます。トンネルエンドポイントにはルーティング可能なインターフェイスがあるので、多くの共通インターフェイス機能を IPsec トンネルに適用できます。

IPsec VTI によって、複数パスの場合のように、物理インターフェイス上における IP ユニキャストおよびマルチキャストの両方の暗号化トラフィックの送受信の柔軟性が高まります。トラフィックは、トンネルインターフェイスから転送されるときに暗号化され、トンネルインターフェイスに転送されると復号化されます。また、IP ルーティング テーブルによって管理され

ます。IP ルーティングを使用してトラフィックをトンネル インターフェイスに転送すると、IPsec VPN 設定が簡単になります。DVTI は他のすべての実際のインターフェイスと同様に機能するため、トンネルがアクティブになるとすぐに Quality of Service（QoS）、ファイアウォール、およびその他のセキュリティ サービスを適用できます。

IPSec VTI に関する詳細については、次の各項を参照してください。

IPsec 仮想トンネル インターフェイスを使用するメリット

IPsec VTI によって、機能を適用できる仮想インターフェイスを設定できます。暗号化されていないテキスト パケットの機能は VTI 上で設定されます。暗号化されたパケットの機能は物理外部インターフェイス上で適用されます。IPsec VTI を使用すると、ネットワーク アドレス変換（NAT）、ACL、QoS などの各種機能のアプリケーションを分離して、それらを暗号化されていないテキストまたは暗号化されたテキスト、あるいはその両方に適用できます。

スタティック VTI（SVTI）と DVTI という 2 つのタイプの VTI インターフェイスが存在します。

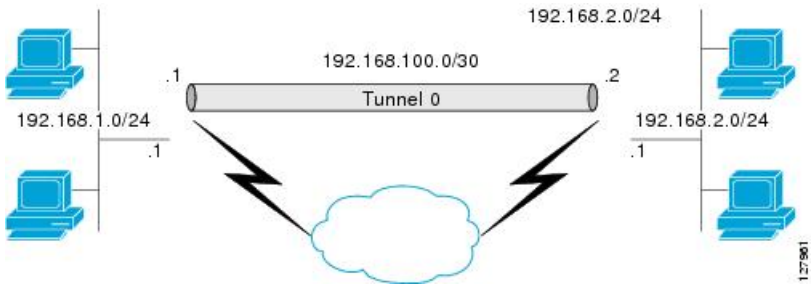
スタティック仮想トンネル インターフェイス

SVTI 設定は、トンネルによって 2 つのサイト間の常にオンであるアクセスが提供される、サイト間接続用に使用できます。

さらに、複数の Cisco IOS ソフトウェア機能を、トンネル インターフェイス上、およびトンネルインターフェイスの物理出力インターフェイス上で直接設定できます。この直接設定によって、ユーザは、暗号化前または暗号化後のパスにおける機能のアプリケーションを確実に管理できます。

次の図に、SVTI の使用方法を示します。

図 *1 : IPsec SVTI*



IPsec VTI によって、ネイティブの IPsec トンネリングがサポートされ、物理インターフェイスのプロパティの大部分が示されます。

SVTI のマルチ SA サポート

デフォルトでは、SVTI のトラフィックセレクタは「any any」に設定されます。その結果、「any any」トラフィックセレクタに対応する SVTI に単一の IPSec SA が接続されます。

Cisco IOS XE Gibraltar 16.12.1 以降では、アクセス制御リスト（ACL）を定義して SVTI に関連付けることで、デフォルトで定義されている「any any」プロキシではなく特定の送信元プロキシと宛先プロキシの間のトラフィックを選択できます。非 any-any トラフィックセレクタごとに IPsec SA が作成されるため、複数の SA が SVTI に接続されます。

この機能は、トンネルモードでの IPSec カプセル化による IPv4 および IPv6 トラフィック保護をサポートしています。この機能は IKEv1 と IKEv2 の両方をサポートしています。

制約事項

- この機能は、共有されたトンネル保護ではサポートされません。
- この機能は、IPSec 混合モードではサポートされません。
- トンネルの両端の SVTI に関連付けられたトラフィックセレクタには、一致する送信元プロキシと宛先プロキシが必要です。トンネルを形成する SVTI のいずれかでトラフィックセレクタを絞り込まないでください。

ACL の特性と SVTI IPSec SA への影響

- SVTI に関連付けられた ACL に「any any」プロキシを含めないでください。「any any」トラフィックセレクタについては、SVTI のデフォルト動作を使用してください（ACL を SVTI に関連付けないでください）。
- SVTI に関連付けられた ACL は **permit** ステートメントのみをサポートしているので、**deny** ステートメントを含めないでください。
- SVTI に関連付けられた ACL の実行時変更はサポートされていません。ACL の ACE を追加または変更する前にトンネルをシャットダウンしてください。
- SVTI への ACL の関連付けを解除すると、既存の IPSec SA が削除され、「IP any any」のデフォルト トラフィック セレクタに関する新しい IPSec SA が形成されます。
- SVTI に関連付けるアクセス制御エントリ（ACE）は 100 までにすることをお勧めします。また、さまざまなトンネルインターフェイスに関連付けられたすべての ACL で使用される ACE の合計が 2000 を超えないようにすることをお勧めします。

逆ルート注入

マルチ SA の SVTI の場合は、IPSec プロファイルで逆ルート注入（RRI）を設定できます。

拡張 ACL または ACE オプション（プロトコル、ポート番号、DHCP など）を使用する場合は、RRI を使用しないでください。ルーティングにはルートマップなどの他の手段を使用してください。

（注） 距離とタグによる RRI 機能は、まだサポートされていません。

SVTI に対するデュアルスタックのサポート

SVTI デュアルスタック機能により、IPv4 を介してトンネリングされる単一の IPsec セキュリティ アソシエーション（SA）を使用して IPv4 トラフィックと IPv6 トラフィックの両方を伝送することが可能になります。IOS XE リリース 17.9 以降では、トンネルインターフェイスの入力側がサードパーティの IPsec クライアントで設定されている場合、ACL の特定のサブネットがサポートされます。また、サードパーティの IPsec クライアントの設定に基づいて、特定のトラフィックセレクタで応答されます。この場合、IPsec は、non-any non-any プロキシ設定をサポートし、トンネルインターフェイスで IPv4 または IPv6 タイプのトラフィックを伝送することを許可します。この機能は、IKEv2 でのみサポートされます。

制約事項

- トンネルモードの設定は、デュアルオーバーレイモードでトンネルインターフェイスを使用する場合に、IPsec プロファイルでのみ許可されます。
- Cisco IOS XE では、ACL フィルタリング インフラストラクチャは、デバイスでローカルに生成されたトラフィックでは機能しません。
- IPsec SA のキー再生成には、一連の同じトラフィックセレクタを使用する必要があります。キー再生成プロセス中にトラフィックセレクタを変更することはできず、変更すると、キー再生成要求はメッセージ *TS_UNACCEPTABLE* をともなって拒否されます。
- IKEv2 レベルでは、最大 16 のトラフィックセレクタが受け入れられます。
- デュアルスタック トンネル インターフェイスの ACL は、サポートされていません。このインターフェイスで設定されている ACL は、デュアルスタック ACL によって上書きされます。

ダイナミック仮想トンネル インターフェイス

DVTI によって、リモートアクセス VPN 用接続のセキュリティ保護とスケーラビリティが向上します。DVTI テクノロジーは、ダイナミック クリプトマップとトンネルを確立するためのダイナミック ハブアンドスポーク方式にとって代わるものです。

（注） IKEv1 または IKEv2 を使用して DVTI を設定できます。レガシー クリプト マップ ベースの設定は、IKEv1 を使用した DVTI しかサポートしません。IKEv2 を使用した DVTI 設定は FlexVPN でのみサポートされます。

DVTI は、サーバと、リモート設定の両方に対して使用可能です。トンネルにより、各 VPN セッションに対して、仮想アクセスインターフェイスがオンデマンドで個別に提供されます。仮想アクセス インターフェイス設定は、仮想テンプレート設定からコピーされます。このコピーには、IPsec 設定と、QoS、NetFlow、ACL といった、仮想テンプレートインターフェイス上で設定されたすべての Cisco IOS ソフトウェア機能が含まれています。

DVTI は、他の現実のインターフェイスと同様に機能するため、トンネルがアクティブになった直後に、QoS、ファイアウォール、またはその他のセキュリティ サービスを適用できます。QoS 機能を使用して、ネットワーク上の各種アプリケーションのパフォーマンスを向上させることが可能です。Cisco IOS ソフトウェア内で提供される各種 QoS 機能の組み合わせを使用して、音声、ビデオ、またはデータ アプリケーションをサポートできます。

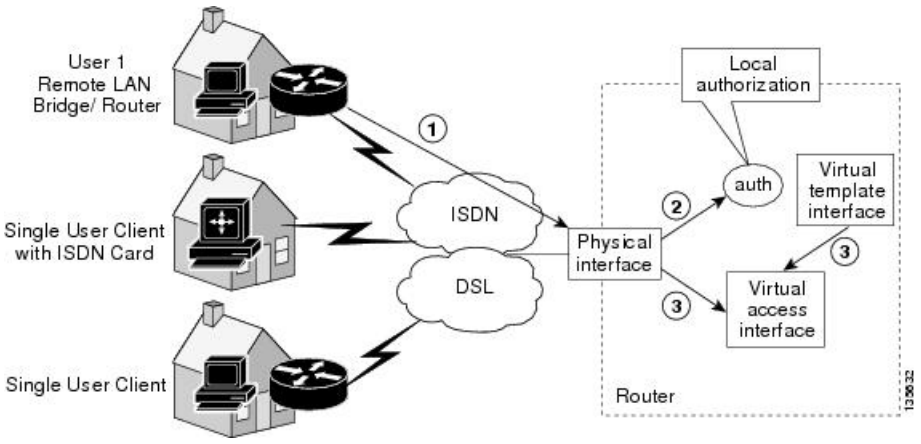
DVTI によって、IP アドレスを効率的に使用できるようになり、また、セキュアな接続を実現できます。DVTI によって、動的にダウンロード可能な、グループごとおよびユーザーごとのポリシーを RADIUS サーバー上で設定できます。グループ単位またはユーザ単位の定義は、拡張認証（Xauth）User または Unity グループを使用して作成することも、証明書から抽出することもできます。DVTI は、標準ベースです。そのため、複数のベンダー環境における相互運用性がサポートされます。IPsec DVTI を使用すれば、リモート アクセス VPN 用のセキュリティ保護が強化された接続を作成できます。また、Cisco Architecture for Voice, Video, and Integrated Data（AVVID）と組み合わせて、IP ネットワーク経由で集約された音声、ビデオ、およびデータを転送できます。DVTI は VPN ルーティングおよび転送（VRF）対応 IPsec の導入を容易にします。VRF は、インターフェイス上で設定されます。

DVTI には、ルータ上での最小限の設定が必要です。単一の仮想テンプレートを設定およびコピーできます。

DVTI によって、IPsec セッション用のインターフェイスが作成され、ダイナミック IPsec VTI の動的なインスタンス化および管理のための仮想テンプレート インフラストラクチャが使用されます。仮想テンプレート インフラストラクチャは、ダイナミック仮想アクセス トンネル インターフェイスを作成するために拡張されます。DVTI は、ハブアンドスポーク設定で使用されます。単一の DVTI で複数のスタティック VTI をサポートできます。

次の図に、DVTI 認証パスを示します。

図 *2:* ダイナミック *IPsec VTI*



上の図の認証は、次のパスに従います。

1. ユーザ 1 がルータを呼び出します。
2. ルータ 1 によって ユーザ 1 が認証されます。

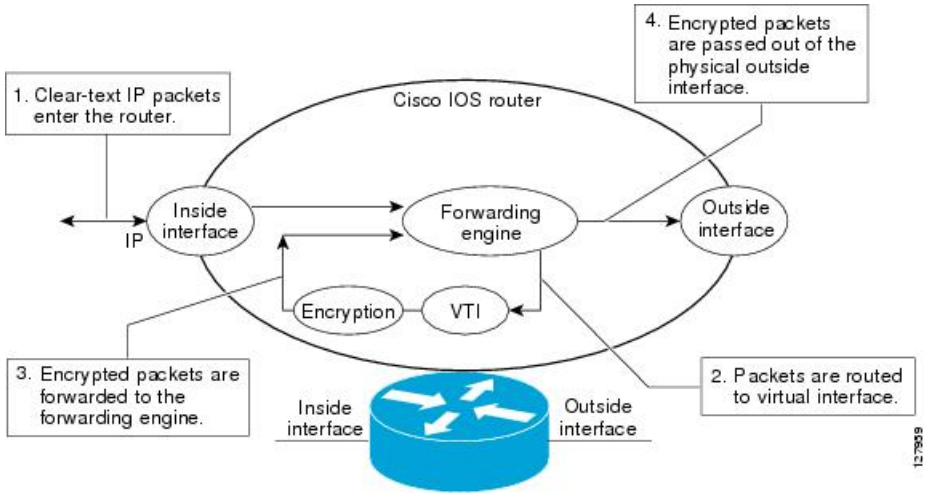
3. IPsec によって、仮想テンプレート インターフェイスから仮想アクセス インターフェイスがコピーされます。

IPsec 仮想トンネル インターフェイスを使用したトラフィックの暗号化

IPsec VTI が設定されると、暗号化がトンネル内で実行されます。トラフィックがトンネル インターフェイスに転送されると、そのトラフィックが暗号化されます。トラフィックの転送は、IP ルーティング テーブルによって処理され、ダイナミックまたはスタティック ルーティングを使用してトラフィックを SVTI にルーティングできます。DVTI では、逆ルート注入が使用されるので、ルーティングの設定がさらに簡単になっています。IP ルーティングを使用してトラフィックを暗号化に転送すると、IPsec VPN 設定が簡単になります。さらに、IPsec 仮想トンネルを使用すれば、IPsec によってマルチキャスト トラフィックを暗号化できます。

次の図に、IPsec トンネルへの IPsec パケット フローを示します。

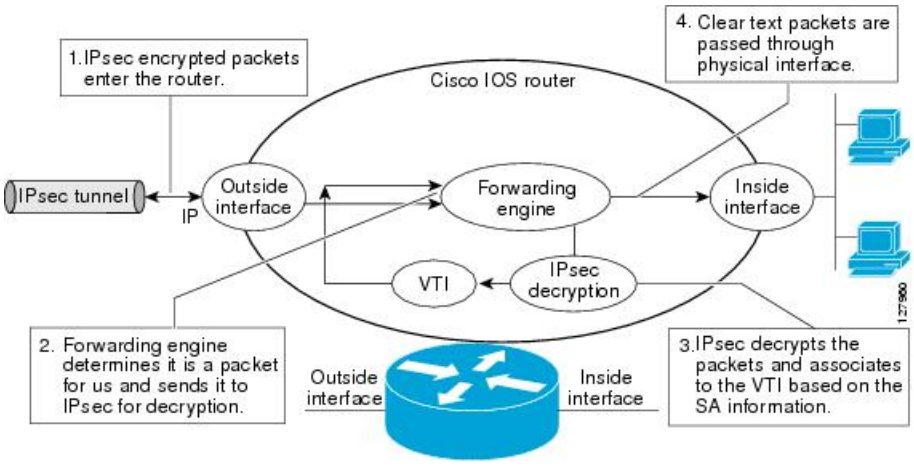
図 *3 : IPsec* トンネルへのパケット フロー



パケットが内部インターフェイスに到着すると、転送エンジンによってパケットが VTI にスイッチングされ、そこで暗号化されます。暗号化されたパケットは転送エンジンに戻され、そこで外部インターフェイスを介してスイッチングされます。

次の図に、IPsec トンネルからのパケット フローを示します。

図 *4: IPsec* トンネルからのパケット フロー



ダイナミック仮想トンネル インターフェイスのライフ サイクル

IPsec プロファイルによって、DVTI のポリシーが定義されます。ダイナミック インターフェイスが、IKE フェーズ 1 および IKE フェーズ 1.5 の終了時に作成されます。ピアに対する IPsec セッションが終了すると、インターフェイスが削除されます。ピアに対する IKE と IPsec SA の両方が削除されると、IPsec セッションが終了します。

IPsec 仮想トンネル インターフェイスを使用したルーティング

VTI はルーティング可能なインターフェイスなので、暗号化プロセスにおけるルーティングの役割は重要です。トラフィックは、VTI の外に転送される場合にだけ暗号化され、VTI に到着するトラフィックは、適宜、復号化およびルーティングされます。VTI を利用すれば、実際のインターフェイスをトンネルエンドポイントとして使用することによって、暗号化トンネルを確立できます。インターフェイスにルーティングしたり、QoS、ファイアウォール、ネットワーク アドレス変換（NAT）、Netflow 統計情報などのサービスを必要に応じて他のインターフェイスに適用したりできます。インターフェイスをモニタして、それにルーティングできます。このインターフェイスは他の Cisco IOS インターフェイスと同様のメリットを提供します。

FlexVPN 混合モードのサポート

FlexVPN 混合モード機能は、IPsec IPv6 トランスポート経由の IPv4 トラフィックの伝送をサポートします。これは、IPsec スタック上でのデュアル スタックのサポートにつながる第 1 段階です。この実装は、IPv4 トラフィックと IPv6 トラフィックの両方に対する単一の IPSec セキュリティ アソシエーション（SA）ペアの使用をサポートしません。

この機能は、IKEv2 とダイナミック VTI を使用したリモート アクセス VPN に対してのみサポートされます。

FlexVPN 混合モード機能は、Cisco IOS XE Everest 16.4.1 からの IPsec IPv4 トランスポート経由の IPv6 トラフィック伝送をサポートします。

IPsec での自動トンネル モードのサポート

複数ベンダー シナリオで VPN ヘッドエンドを設定する場合は、ピアまたはレスポンダの技術的な詳細を認識しておく必要があります。たとえば、一部のデバイスは IPsec トンネルを使用しているが、他のデバイスは Generic Routing Encapsulation（GRE）または IPsec トンネルを使用している場合やトンネルが IPv4 または IPv6 の場合があります。最後のケースでは、インターネット キー エクスチェンジ（IKE）プロファイルと仮想テンプレートを設定する必要があります。

トンネル モード自動選択機能は、設定を容易にし、レスポンダの詳細の入手を支援します。この機能は、IKE プロファイルから仮想アクセス インターフェイスが作成されるとすぐに、トンネリング プロトコル（GRE または IPsec）とトランスポート プロトコル（IPv4 または IPv6）を自動的に仮想テンプレートに適用します。この機能は、Cisco AnyConnect VPN Client や Microsoft Windows7 Client などのマルチベンダー リモート アクセスを集約しているデュアル スタック ハブ上で役に立ちます。

（注） トンネル モード自動選択機能は、レスポンダの設定のみを容易にします。トンネルはイニシエータに対して静的に設定する必要があります。

VTI に対する IPSec 混合モードのサポート

IPSec 混合モード機能は、IPsec IPv6 トランスポート経由の IPv4 トラフィックの伝送をサポートします。これは、IPsec スタック上でのデュアル スタックのサポートにつながる第 1 段階です。この実装は、IPv4 トラフィックと IPv6 トラフィックの両方に対する単一の IPSec セキュリティ アソシエーション（SA）ペアの使用をサポートしません。

この機能は、SVTI、DVTI、IKEv1、および IKEv2 でサポートされます。

IPsec 仮想トンネル インターフェイスの設定方法

スタティック IPsec 仮想トンネル インターフェイスの設定

始める前に

IPsec プロファイルのトンネル保護を設定する前に、トンネルインターフェイスをシャットダウンする必要があります。設定後、トンネルインターフェイスを手動で有効にしてください。

手順の概要

1. **enable**
2. **configure terminal**
3. **crypto IPsec profile** *profile-name*

4. **set transform-set** *transform-set-name* [*transform-set-name2...transform-set-name6*]
5. **exit**
6. **interface** *type number*
7. **ip address** *address mask*
8. **tunnel mode ipsec ipv4**
9. **tunnel source** *interface-type interface-number*
10. **tunnel destination** *ip-address*
11. **tunnel protection IPsec profile** *profile-name*
12. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# 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 3	**crypto IPsec profile** *profile-name* 例： `Device(config)# crypto IPsec profile PROF`	2 つの IPsec デバイス間の IPsec 暗号化に使用される IPsec パラメータを定義して、IPsec プロファイル コンフィギュレーション モードを開始します。
ステップ 4	**set transform-set** *transform-set-name* [*transform-set-name2...transform-set-name6*] 例： `Device(ipsec-profile)# set transform-set tset`	使用可能なトランスフォームセットを指定します。
ステップ 5	**exit** 例： `Device(ipsec-profile)# exit`	IPsecプロファイル コンフィギュレーション モードを終了して、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 6	**interface** *type number* 例： `Device(config)# interface tunnel 0`	トンネルが設定されるインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 7	**ip address** *address mask* 例：	IP アドレスおよびマスクを指定します。

	コマンドまたはアクション	目的
	`Device(config-if)# ip address 10.1.1.1 255.255.255.0`	
ステップ 8	**tunnel mode ipsec ipv4** 例： `Device(config-if)# tunnel mode ipsec ipv4`	トンネルのモードを定義します。
ステップ 9	**tunnel source** *interface-type interface-number* 例： `Device(config-if)# tunnel source loopback 0`	トンネルの送信元をループバック インターフェイスとして指定します。* （注） * 仮想テンプレートを使用してトンネルモード自動選択機能を設定する場合は、interface virtual-template number type tunnel コマンドでトンネル送信元とトンネルモードを省略します。トンネル送信元とトンネルモードが指定されている場合、IPv6 トランスポートを使用するクライアントは接続に失敗します。
ステップ 10	**tunnel destination** *ip-address* 例： `Device(config-if)# tunnel destination 172.16.1.1`	トンネルの宛先の IP アドレスを指定します。
ステップ 11	**tunnel protection IPsec profile** *profile-name* 例： `Device(config-if)# tunnel protection IPsec profile PROF`	トンネル インターフェイスを IPsec プロファイルに関連付けます。
ステップ 12	**end** 例： `Device(config-if)# end`	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、特権 EXEC モードに戻ります。

IPsec 仮想トンネルインターフェイスを介した BGP の設定

必要に応じて、2つのルータの仮想トンネルインターフェイスを介してBGPを設定するには、次の作業を実行します。

始める前に

スタティック IPsec 仮想トンネル インターフェイスの設定 （9 ページ） の手順を実行します。

手順の概要

1. **router bgp** *autonomous-system-number*
2. **neighbor** *ip-address* **remote-as** *autonomous-system-number*
3. **network** *network-ip-address* **mask** *subnet-mask*
4. **exit**
5. 2 番目のルータで次のコマンドを入力します。
6. **router bgp** *autonomous-system-number*
7. **neighbor** *ip-address* **remote-as** *autonomous-system-number*
8. **network** *network-ip-address* **mask** *subnet-mask*

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**router bgp** *autonomous-system-number* 例： `Device(config)# router bgp 65510`	ルータ コンフィギュレーションモードを開始して、BGP ルーティング プロセスを作成します。 *autonomous-system-number*：他の BGP ルータに対するルータを指定し、同時に渡されるルーティング情報のタギングをする自律システムの番号。番号の範囲は 1 ～ 65535 です。 この例では、この手順の最初のルータは「65510」として識別されます。
ステップ 2	**neighbor** *ip-address* **remote-as** *autonomous-system-number* 例： `Device(config-router)# neighbor 10.1.1.2 remote-as 65511`	*ip-address*：隣接ルータのトンネルインターフェイスの IP アドレス。 *autonomous-system-number*：2 番目のルータのルータを識別する自律システムの番号。番号の範囲は 1 ～ 65535 です。
ステップ 3	**network** *network-ip-address* **mask** *subnet-mask* 例： `Device(config-router)# network 2.2.2.0 mask 255.255.255.0`	*network-ip-address*：BGP でアドバタイズされるネットワークの IP アドレス。たとえば、ループバック インターフェイスの IP アドレスです。 *subnet-mask*：BGP でアドバタイズされるネットワークのサブネットマスク。 （注） BGP ネットワークコマンドの network および mask は、BGP に取り込まれて BGP ネイバーにアドバタイズされるように、ルーティングテーブルにすでに存在するルートと正確に一致する必要があります。これは、network ステートメントがインターフェイス ネットワークを「カバーする」だけで、インターフェイスからマスクを使用してネットワークを取得する EIGRP、OSPF とは異なります。

	コマンドまたはアクション	目的
ステップ 4	**exit** 例： `Device(config-router)# exit`	ルータ コンフィギュレーション モードを終了します。
ステップ 5	2 番目のルータで次のコマンドを入力します。	
ステップ 6	**router bgp** *autonomous-system-number* 例： `Device(config)# router bgp 65511`	ルータ コンフィギュレーション モードを開始して、BGP ルーティング プロセスを作成します。 *autonomous-system-number*：他の BGP ルータに対するルータを指定し、同時に渡されるルーティング情報のタギングをする自律システムの番号。番号の範囲は 1 ～ 65535 です。 この例では、この手順の 2 番目のルータは「65511」として識別されます。
ステップ 7	**neighbor** *ip-address* **remote-as** *autonomous-system-number* 例： `Device(config-router)# neighbor 10.1.1.1 remote-as 65510`	*ip-address*：隣接ルータのトンネルインターフェイスの IP アドレス。
ステップ 8	**network** *network-ip-address* **mask** *subnet-mask* 例： `Device(config-router)# network 1.1.1.0 mask 255.255.255.0`	*network-ip-address*：BGP でアドバタイズされるネットワークの IP アドレス。たとえば、ループバック インターフェイスの IP アドレスです。 *subnet-mask*：BGP でアドバタイズされるネットワークのサブネットマスク。 （注） 正確なネットワーク IP アドレスおよびサブネットマスクを使用してください。

ダイナミック IPsec 仮想トンネル インターフェイスの設定

手順の概要

1. **enable**
2. **configure terminal**
3. **crypto ipsec profile** *profile-name*
4. **set transform-set** *transform-set-name* [*transform-set-name2...transform-set-name6*]
5. **exit**
6. **interface virtual-template** *number* **type tunnel**
7. **tunnel mode ipsec ipv4**
8. **tunnel protection IPsec profile** *profile-name*

9. **exit**
10. **crypto isakamp profile** *profile-name*
11. **match identity address** *ip-address mask*
12. **virtual template** *template-number*
13. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# 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 3	**crypto ipsec profile** *profile-name* 例： `Device(config)# crypto ipsec profile PROF`	2 つの IPsec デバイス間の IPsec 暗号化に使用される IPsec パラメータを定義して、IPsec プロファイル コンフィギュレーション モードを開始します。
ステップ 4	**set transform-set** *transform-set-name* [*transform-set-name2...transform-set-name6*] 例： `Device(ipsec-profile)# set transform-set tset`	クリプト マップ エントリで使用可能なトランスフォーム セットを指定します。
ステップ 5	**exit** 例： `Device(ipsec-profile)# exit`	IPsec プロファイル コンフィギュレーション モードを終了して、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 6	**interface virtual-template** *number* **type tunnel** 例： `Device(config)# interface virtual-template 2 type tunnel`	仮想テンプレート トンネル インターフェイスを定義し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 7	**tunnel mode ipsec ipv4** 例： `Device(config-if)# tunnel mode ipsec ipv4`	トンネルのモードを定義します。
ステップ 8	**tunnel protection IPsec profile** *profile-name* 例： `Device(config-if)# tunnel protection ipsec profile PROF`	トンネル インターフェイスを IPsec プロファイルに関連付けます。

	コマンドまたはアクション	目的
ステップ 9	**exit** 例： `Device(config-if)# exit`	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。
ステップ 10	**crypto isakamp profile** *profile-name* 例： `Device(config)# crypto isakamp profile profile1`	仮想テンプレートに使用される ISAKAMP プロファイルを定義します。
ステップ 11	**match identity address** *ip-address mask* 例： `Device(conf-isa-prof)# match identity address 10.1.1.0 255.255.255.0`	ISAKMP プロファイルからの ID を照合して、isakmp-profile コンフィギュレーション モードを開始します。
ステップ 12	**virtual template** *template-number* 例： `Device(config)# virtual-template 1`	ISAKAMP プロファイルにアタッチされた仮想テンプレートを指定します。
ステップ 13	**end** 例： `Device(config)# end`	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードを開始します。

IKEv1 を使用したダイナミック仮想トンネル インターフェイスのマルチ SA サポートの設定

（注） セキュリティに対する脅威は、脅威からの保護に役立つ暗号化技術と同様に絶え間なく変化しています。最新のシスコの暗号化に関する推奨事項については、『Next Generation Encryption』（NGE）ホワイト ペーパーを参照してください。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip vrf** *vrf-name*
4. **rd** *route-distinguisher*
5. **exit**
6. **crypto keyring** *keyring-name*
7. **pre-shared-key** *address* **key** *key*
8. **exit**
9. **crypto isakmp profile** *profile-name*
10. **keyring** *keyring-name*

11. **match identity** *address mask*
12. **virtual-template** *template-number*
13. **exit**
14. **crypto ipsec transform-set** *transform-set-name transform1* [*transform2*] [*transform3*]
15. **exit**
16. **crypto ipsec profile** *name*
17. **set security-policy limit** *maximum-limit*
18. **set transform-set** *transform-set-name* [*transform-set-name2 .... transform-set-name6*]
19. **exit**
20. **interface virtual-template** *number type tunnel*
21. **ip vrf forwarding** *vrf-name*
22. **ip unnumbered** *type number*
23. **tunnel mode ipsec ipv4**
24. **tunnel protection profile ipsec** *profile-name*
25. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。
ステップ 2	**configure terminal** 例： `Device# 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 3	**ip vrf** *vrf-name* 例： `Device(config)# ip vrf VRF-100-1`	VRFインスタンスを定義し、VRFコンフィギュレーション モードを開始します。
ステップ 4	**rd** *route-distinguisher* 例： `Device(config-vrf)# rd 100:21`	VRF のルーティング テーブルと転送テーブルを作成します。
ステップ 5	**exit** 例： `Device(config-vrf)# exit`	VRF コンフィギュレーション モードを終了して、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 6	**crypto keyring** *keyring-name* 例： `Device(config)# crypto keyring cisco-100-1`	暗号キーリングを定義し、キーリング コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 7	**pre-shared-key** *address* **key** *key* 例： `Device(config-keyring)# pre-shared-key address 10.1.1.1 key cisco-100-1`	インターネット キー エクスチェンジ（IKE）認証に使用する事前共有キーを定義します。
ステップ 8	**exit** 例： `Device(config-keyring)# exit`	キーリング コンフィギュレーション モードを終了して、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 9	**crypto isakmp profile** *profile-name* 例： `Device(config)# crypto isakmp profile cisco-isakmp-profile-100-1`	ISAKMP プロファイルを定義し、ISAKMP コンフィギュレーション モードを開始します。
ステップ 10	**keyring** *keyring-name* 例： `Device(conf-isa-prof)# keyring cisco-100-1`	ISAKMP モードでキーリングを設定します。
ステップ 11	**match identity** *address mask* 例： `Device(conf-isa-prof)# match identity address 10.1.1.0 255.255.255.0`	ISAKMP プロファイルからの ID を照合します。
ステップ 12	**virtual-template** *template-number* 例： `Device(conf-isa-prof)# virtual-template 101`	仮想アクセス インターフェイスの複製に使用される仮想テンプレートを指定します。
ステップ 13	**exit** 例： `Device(conf-isa-prof)# exit`	ISAKMP プロファイル コンフィギュレーション モードを終了して、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 14	**crypto ipsec transform-set** *transform-set-name transform1* [*transform2*] [*transform3*] 例： `Device(config)# crypto ipsec transform-set cisco esp-aes esp-sha-hmac`	トランスフォーム セットを定義し、暗号トランスフォーム コンフィギュレーション モードを開始します。
ステップ 15	**exit** 例： `Device(conf-crypto-trans)# exit`	クリプト トランスフォーム コンフィギュレーションモードを終了して、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 16	**crypto ipsec profile** *name* 例： `Device(config)# crypto ipsec profile cisco-ipsec-profile-101`	2 つの IPsec デバイス間の IPsec 暗号化に使用される IPsec パラメータを定義して、IPsec プロファイル コンフィギュレーション モードを開始します。

	コマンドまたはアクション	目的
ステップ 17	**set security-policy limit** *maximum-limit* 例： `Device(ipsec-profile)# set security-policy limit 3`	仮想アクセス インターフェイスごとに作成可能なフロー数の上限を定義します。
ステップ 18	**set transform-set** *transform-set-name* [*transform-set-name2 .... transform-set-name6*] 例： `Device(ipsec-profile)# set transform-set cisco`	クリプト マップ エントリで使用されるトランスフォーム セットを指定します。
ステップ 19	**exit** 例： `Device(ipsec-profile)# exit`	IPsec プロファイル コンフィギュレーション モードを終了して、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 20	**interface virtual-template** *number type tunnel* 例： `Device(config)# interface virtual-template 101 type tunnel`	インターフェイスを設定可能な仮想テンプレート インターフェイスを作成し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 21	**ip vrf forwarding** *vrf-name* 例： `Device(config-if)# ip vrf forwarding VRF-100-1`	VRF インスタンスと仮想テンプレート インターフェイスを関連付けます。
ステップ 22	**ip unnumbered** *type number* 例： `Device(config-if)# ip unnumbered GigabitEthernet 0.0`	明示的な IP アドレスをインターフェイスに割り当てずにインターフェイス上の IP 処理をイネーブルにします。
ステップ 23	**tunnel mode ipsec ipv4** 例： `Device(config-if)# tunnel mode ipsec ipv4`	トンネルのモードを定義します。
ステップ 24	**tunnel protection profile ipsec** *profile-name* 例： `Device(config-if)# tunnel protection ipsec profile PROF`	トンネル インターフェイスを IPsec プロファイルに関連付けます。
ステップ 25	**end** 例： `Device(config-if)# end`	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、特権 EXEC モードに戻ります。

SVTI に対する IPsec 混合モードのサポートの設定

手順の概要

1. **enable**
2. **configure terminal**
3. **crypto IPsec profile** *profile-name*
4. **set transform-set** *transform-set-name* [*transform-set-name2...transform-set-name6*]
5. **exit**
6. **interface** *type number*
7. **ip address** *address mask*
8. 次のいずれかを実行します。
   - **tunnel mode ipsec ipv4 v6-overlay**
   - **tunnel mode ipsec ipv6 v4-overlay**
9. **tunnel source** *interface-type interface-type*
10. **tunnel destination** *ip-address*
11. **tunnel protection IPsec profile** *profile-name*
12. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# 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 3	**crypto IPsec profile** *profile-name* 例： `Device(config)# crypto IPsec profile PROF`	2 つの IPsec デバイス間の IPsec 暗号化に使用される IPsec パラメータを定義して、IPsec プロファイル コンフィギュレーション モードを開始します。
ステップ 4	**set transform-set** *transform-set-name* [*transform-set-name2...transform-set-name6*] 例： `Device(ipsec-profile)# set transform-set tset`	クリプト マップ エントリで使用可能なトランスフォーム セットを指定します。

	コマンドまたはアクション	目的
ステップ 5	**exit** 例： `Device(ipsec-profile)# exit`	IPsecプロファイル コンフィギュレーション モードを終了して、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 6	**interface** *type number* 例： `Device(config)# interface tunnel 0`	トンネルが設定されるインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 7	**ip address** *address mask* 例： `Device(config-if)# ip address 10.1.1.1 255.255.255.0`	IP アドレスおよびマスクを指定します。
ステップ 8	次のいずれかを実行します。 • **tunnel mode ipsec ipv4 v6-overlay** • **tunnel mode ipsec ipv6 v4-overlay** 例： `Device(config-if)# tunnel mode ipsec ipv4 v6-overlay`	トンネルのモードを定義します。
ステップ 9	**tunnel source** *interface-type interface-type* 例： `Device(config-if)# tunnel source loopback 0`	トンネルの送信元をループバック インターフェイスとして指定します。
ステップ 10	**tunnel destination** *ip-address* 例： `Device(config-if)# tunnel destination 172.16.1.1`	トンネルの宛先の IP アドレスを指定します。
ステップ 11	**tunnel protection IPsec profile** *profile-name* 例： `Device(config-if)# tunnel protection IPsec profile PROF`	トンネルインターフェイスをIPsecプロファイルに関連付けます。
ステップ 12	**end** 例： `Device(config-if)# end`	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、特権 EXEC モードに戻ります。

ダイナミック VTI に対する IPsec 混合モードのサポートの設定

手順の概要

1. **enable**
2. **configure terminal**
3. **crypto ipsec profile** *profile-name*
4. **set mixed mode**
5. **set transform-set** *transform-set-name* [*transform-set-name2...transform-set-name6*]
6. **exit**
7. **interface virtual-template** *number* **type tunnel**
8. **tunnel mode ipsec ipv4**
9. **tunnel protection IPsec profile** *profile-name*
10. **exit**
11. **crypto isakamp profile** *profile-name*
12. **match identity address** *ip-address mask*
13. **virtual template** *template-number*
14. **end**

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	**enable** 例： `Device> enable`	特権 EXEC モードを有効にします。 • パスワードを入力します（要求された場合）。
ステップ 2	**configure terminal** 例： `Device# 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 3	**crypto ipsec profile** *profile-name* 例： `Device(config)# crypto ipsec profile PROF`	2 つの IPsec デバイス間の IPsec 暗号化に使用される IPsec パラメータを定義して、IPsec プロファイル コンフィギュレーション モードを開始します。
ステップ 4	**set mixed mode** 例： `Device(config)# set mixed mode`	2 つの IPsec デバイス間の IPsec 暗号化に使用される IPsec パラメータを定義して、IPsec プロファイル コンフィギュレーション モードを開始します。
ステップ 5	**set transform-set** *transform-set-name* [*transform-set-name2...transform-set-name6*] 例： `Device(ipsec-profile)# set transform-set tset`	クリプト マップ エントリで使用可能なトランスフォーム セットを指定します。

	コマンドまたはアクション	目的
ステップ 6	**exit** 例： `Device(ipsec-profile)# exit`	IPsecプロファイル コンフィギュレーション モードを終了して、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 7	**interface virtual-template** *number* **type tunnel** 例： `Device(config)# interface virtual-template 2 type tunnel`	仮想テンプレート トンネル インターフェイスを定義し、インターフェイス コンフィギュレーション モードを開始します。
ステップ 8	**tunnel mode ipsec ipv4** 例： `Device(config-if)# tunnel mode ipsec ipv4`	トンネルのモードを定義します。
ステップ 9	**tunnel protection IPsec profile** *profile-name* 例： `Device(config-if)# tunnel protection ipsec profile PROF`	トンネルインターフェイスをIPsecプロファイルに関連付けます。
ステップ 10	**exit** 例： `Device(config-if)# exit`	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。
ステップ 11	**crypto isakamp profile** *profile-name* 例： `Device(config)# crypto isakamp profile profile1`	仮想テンプレートに使用されるISAKAMPプロファイルを定義します。
ステップ 12	**match identity address** *ip-address mask* 例： `Device(conf-isa-prof)# match identity address 10.1.1.0 255.255.255.0`	ISAKMP プロファイルからの ID を照合して、isakmp-profile コンフィギュレーション モードを開始します。
ステップ 13	**virtual template** *template-number* 例： `Device(config)# virtual-template 1`	ISAKAMPプロファイルにアタッチされた仮想テンプレートを指定します。
ステップ 14	**end** 例： `Device(config)# end`	グローバル コンフィギュレーション モードを終了し、特権 EXEC モードを開始します。

スタティック IPsec 仮想トンネルインターフェイスのマルチ SA サポートの設定

ステップ 1 **enable**

例：

```
Device> enable
```

特権 EXEC モードを有効にします。パスワードを入力します（要求された場合）。

ステップ 2 **configure terminal**

例：

```
Device# 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 3 **crypto IPsec profile** *profile-name*

例：

```
Device(config)# crypto IPsec profile PROF
```

2 つの IPsec デバイス間の IPsec 暗号化に使用される IPsec パラメータを定義して、IPsec プロファイル コンフィギュレーション モードを開始します。

ステップ 4 **set transform-set** *transform-set-name* [*transform-set-name2...transform-set-name6*]

例：

```
Device(ipsec-profile)# set transform-set tset
```

使用可能なトランスフォームセットを指定します。

ステップ 5 **exit**

例：

```
Device(ipsec-profile)# exit
```

IPsec プロファイル コンフィギュレーション モードを終了して、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 6 **interface** *type number*

例：

```
Device(config)# interface tunnel 0
```

トンネルが設定されるインターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。

ステップ 7 **ip address** *address mask*

例：

```
Device(config-if)# ip address 10.1.1.1 255.255.255.0
```

IP アドレスおよびマスクを指定します。

ステップ 8 **tunnel mode ipsec** {**ipv4** | **ipv6**}

例：

```
Device(config-if)# tunnel mode ipsec ipv4
```

トンネルのモードを定義します。

ステップ 9 **tunnel source** *interface-type interface-number*

例：

```
Device(config-if)# tunnel source loopback 0
```

トンネルの送信元をループバック インターフェイスとして指定します。

ステップ 10 **tunnel destination** *ip-address*

例：

```
Device(config-if)# tunnel destination 172.16.1.1
```

トンネルの宛先の IP アドレスを指定します。

ステップ 11 **tunnel protection ipsec policy** {**ipv4** | **ipv6**} *acl*

例：

```
Device(config-if)# tunnel protection ipsec policy ipv4 ipsec-acl1
```

ACL を SVTI に関連付けて、非 any-any トラフィックセレクタを定義します。

ステップ 12 **tunnel protection ipsec profile** *profile-name*

例：

```
Device(config-if)# tunnel protection IPsec profile PROF
```

トンネル インターフェイスを IPsec プロファイルに関連付けます。

ステップ 13 **exit**

例：

```
Device(config-if)# exit
```

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、グローバル コンフィギュレーション モードに入ります。

ステップ 14 **ip access-list extended** *name* または **ipv6 access-list** *name*

例：

IPv4：

```
Device(config)# ip access-list extended ipsec-acl1
```

IPv6：

```
Device(config)# ipv6 access-list ipsec-acl1
```

名前を使用して拡張IPアクセス リストを定義し、拡張名前付きアクセス リストのコンフィギュレーション モードを開始します。

ステップ 15 **permit** *protocol source* [*source-wildcard*] *destination* [*destination-wildcard*] [**option** *option-name*]

例：

```
Device(config-ext-nacl)# permit ip 30.0.1.0 0.0.0.255 10.0.1.0 0.0.0.255
```

ステートメントに指定されたすべての条件に一致するトラフィックを許可します。

送信元プロキシと宛先プロキシの両方にキーワード**any**をワイルドカードとして使用しないでください。「any any」トラフィックセレクタの場合は、ACL が関連付けられていないデフォルトの SVTI を使用します。

deny ステートメントは使用しないでください。

ステップ 16 **end**

例：

```
Device(config-ext-nacl)# end
```

標準の名前付きアクセス リスト コンフィギュレーション モードを終了し、特権 EXEC モードを開始します。

デュアルオーバーレイとしてのトンネルモードの設定

トンネルモードをデュアルオーバーレイとして設定するには、次の手順を実行します。

ステップ 1 **enable**

例：

```
Device> enable
```

特権 EXEC モードを有効にします。パスワードを入力します（要求された場合）。

ステップ 2 **configure terminal**

例：

```
Device# 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 3 **interface tunnel** *type number*

例：

```
Device(config)# interface tunnel 1
```

トンネルインターフェイスおよび番号を指定し、インターフェイス コンフィギュレーション モードを開始します。

ステップ 4 **ipv6 enable**

例：

```
Device(config-if)# ipv6 enable
```

明示的なIPv6アドレスが設定されていないインターフェイスにおけるIPv6処理をイネーブルにします。

ステップ5 **tunnel source** { **ipv4-address | interface-type | interface-number**}

例：

```
Device(config-if)# tunnel source Gigabitethernet 1
```

送信元 IPv6 アドレスまたは送信元インターフェイス タイプおよびトンネル インターフェイスの番号を指定します。インターフェイスのタイプと番号が指定されている場合、そのインターフェイスはIPv6アドレスを使用して設定する必要があります。

ステップ6 **tunnel mode ipsec dual-overlay**

例：

```
Device(config-if)# tunnel mode ipsec dual-overlay
```

デュアルオーバーレイ トンネルを指定します。**tunnel mode ipsec dual-overlay** コマンドは、トンネルのカプセル化プロトコルを指定します。

ステップ7 **tunnel destination ip address** *address mask*

例：

```
Device(config-if)# tunnel destination 89.89.89.1 255.255.255.255.0
```

トンネル インターフェイスの宛先 IPv6 アドレスを指定します。

ステップ8 **tunnel protection ipsec profile** *ipserc profile-name*

例：

```
Device(config-if)# tunnel protection IPsec profile ipsecprof
```

トンネルインターフェイスをIPsecプロファイルに関連付けます。*name*引数には、IPsecプロファイルの名前を指定します。この値は、**crypto IPsec profile** *name* コマンドで指定した *name* と一致する必要があります。

ステップ9 **exit**

例：

```
Device(config-if)# exit
```

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、グローバル コンフィギュレーション モードに入ります。

ステップ10 **end**

例：

```
Device(config-if)# end
```

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、特権 EXEC モードに戻ります。

IPsec 仮想トンネル インターフェイスの設定例

例：IPsec を使用したスタティック仮想トンネル インターフェイス

次の設定例では、ピア間の認証用に事前共有キーが使用されています。VPN トラフィックは、暗号化のために IPsec VTI に転送されてから、物理インターフェイスに送信されます。サブネット 10 のトンネルでは、IPsec ポリシーに関してパケットがチェックされ、IPsec 暗号化のために暗号エンジン（CE）に渡されます。次の図に、IPSec VTI 設定を示しています。

図 *5 : IPsec* を使用した *VTI*

ルータのコンフィギュレーション

```
version 12.3
service timestamps debug datetime
service timestamps log datetime
hostname 7200-3
no aaa new-model
ip subnet-zero
ip cef
controller ISA 6/1
!
crypto isakmp policy 1
encr aes
authentication pre-share
group 14
crypto isakmp key Cisco12345 address 0.0.0.0 0.0.0.0
crypto ipsec transform-set T1 esp-aes esp-sha-hmac
crypto ipsec profile P1
set transform-set T1
!
interface Tunnel0
 ip address 10.0.51.203 255.255.255.0

 load-interval 30
 tunnel source 10.0.149.203
 tunnel destination 10.0.149.217
 tunnel mode IPsec ipv4
 tunnel protection IPsec profile P1
!

 ip address 10.0.149.203 255.255.255.0
 duplex full
!

 ip address 10.0.35.203 255.255.255.0
 duplex full
!
ip classless
ip route 10.0.36.0 255.255.255.0 Tunnel0
line con 0
line aux 0
line vty 0 4
end
```

ルータのコンフィギュレーション

```
version 12.3
hostname c1750-17
no aaa new-model
ip subnet-zero
ip cef
crypto isakmp policy 1
encr aes
authentication pre-share
group 14
crypto isakmp key Cisco12345 address 0.0.0.0 0.0.0.0
crypto ipsec transform-set T1 esp-aes esp-sha-hmac
crypto ipsec profile P1
set transform-set T1
!
interface Tunnel0
 ip address 10.0.51.217 255.255.255.0

 tunnel source 10.0.149.217
 tunnel destination 10.0.149.203
 tunnel mode ipsec ipv4
 tunnel protection ipsec profile P1
!
interface
 ip address 10.0.149.217 255.255.255.0
 speed 100
 full-duplex
!
interface
 ip address 10.0.36.217 255.255.255.0
 load-interval 30
 full-duplex
!
ip classless
ip route 10.0.35.0 255.255.255.0 Tunnel0
line con 0
line aux 0
line vty 0 4
end
```

例：IPsec スタティック仮想トンネル インターフェイスの結果の確認

ここでは、設定が正しく動作しているか確認するうえで利用可能な情報を示します。次の出力では、Tunnel 0 およびライン プロトコルが「up」状態です。ライン プロトコルが「down」状態の場合、セッションは非アクティブです。

IPsec スタティック仮想トンネル インターフェイスの確認

```
Router# show interface tunnel 0

Tunnel0 is up, line protocol is up
Hardware is Tunnel
Internet address is 10.0.51.203/24
MTU 1514 bytes, BW 9 Kbit, DLY 500000 usec,
reliability 255/255, txload 103/255, rxload 110/255
Encapsulation TUNNEL, loopback not set
Keepalive not set
Tunnel source 10.0.149.203, destination 10.0.149.217
Tunnel protocol/transport ipsec/ip, key disabled, sequencing disabled
```

```
Tunnel TTL 255
Checksumming of packets disabled, fast tunneling enabled
Tunnel transmit bandwidth 8000 (kbps)
Tunnel receive bandwidth 8000 (kbps)
Tunnel protection via IPsec (profile "P1")
Last input never, output never, output hang never
Last clearing of "show interface" counters never
Input queue: 1/75/0/0 (size/max/drops/flushes); Total output drops: 0
Queueing strategy: fifo
Output queue: 0/0 (size/max)
30 second input rate 13000 bits/sec, 34 packets/sec
30 second output rate 36000 bits/sec, 34 packets/sec
191320 packets input, 30129126 bytes, 0 no buffer
Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
59968 packets output, 15369696 bytes, 0 underruns
0 output errors, 0 collisions, 0 interface resets
0 output buffer failures, 0 output buffers swapped out

Router# show crypto session

Crypto session current status
Interface: Tunnel0
Session status: UP-ACTIVE
Peer: 10.0.149.217 port 500
IKE SA: local 10.0.149.203/500 remote 10.0.149.217/500 Active
IPsec FLOW: permit ip 0.0.0.0/0.0.0.0 0.0.0.0/0.0.0.0
Active SAs: 4,
Router# show ip route

Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
ia - IS-IS inter area, * - candidate default, U - per-user static route
o - ODR, P - periodic downloaded static route
Gateway of last resort is not set
10.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
C 10.0.35.0/24 is directly connected, Ethernet3/3
S 10.0.36.0/24 is directly connected, Tunnel0
C 10.0.51.0/24 is directly connected, Tunnel0
C 10.0.149.0/24 is directly connected, Ethernet3/0
```

例：VRF 認識スタティック仮想トンネル インターフェイス

VRF をスタティック VTI の例に追加するには、次の例で示すように、**ipvrf** コマンドおよび **ip vrf forwarding** コマンドを設定に含めます。

C8000 ルータ設定

```
hostname c8000
.
.
ip vrf sample-vti1
 rd 1:1
 route-target export 1:1
 route-target import 1:1
!
.
```

```
.
interface Tunnel0
 ip vrf forwarding sample-vti1
 ip address 10.0.51.217 255.255.255.0
 tunnel source 10.0.149.217
 tunnel destination 10.0.149.203
 tunnel mode ipsec ipv4
 tunnel protection ipsec profile P1
.
.
!
end
```

例：QoS を使用したスタティック仮想トンネル インターフェイス

トンネルインターフェイスの下に**service-policy** ステートメントを指定することによって、QoS ポリシーをトンネルエンドポイントに適用できます。次に、トンネルインターフェイスからトラフィックをポリシングする例を示します。

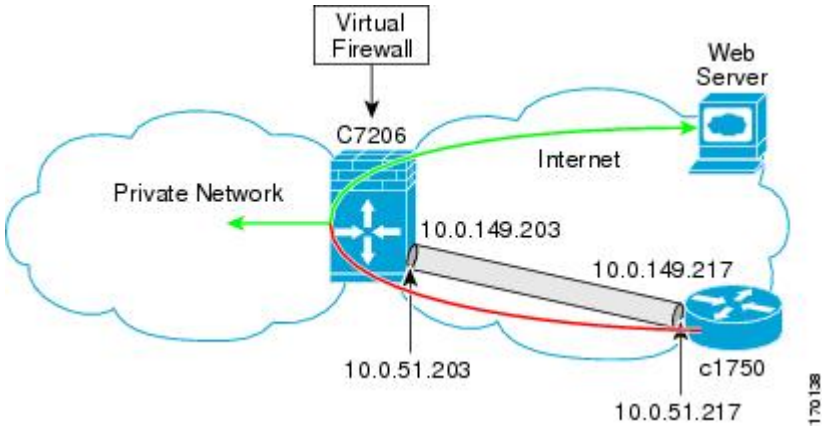
C8000 ルータ設定

```
hostname c8000
.
.
class-map match-all VTI
 match any
!
policy-map VTI
  class VTI
  police cir 2000000
    conform-action transmit
    exceed-action drop
!
.
.
interface Tunnel0
 ip address 10.0.51.217 255.255.255.0
 tunnel source 10.0.149.217
 tunnel destination 10.0.149.203
 tunnel mode ipsec ipv4
 tunnel protection ipsec profile P1
 service-policy output VTI
!
.
.
!
end
```

例：仮想ファイアウォールを使用したスタティック仮想トンネルインターフェイス

仮想ファイアウォールを SVTI トンネルに適用することによって、スポークからのトラフィックを、ハブを通過させてインターネットに送信できます。次の図に、企業ファイアウォールによって本質的に保護されているスポークを使用した SVTI を示します。

図 *6 :* 仮想ファイアウォールを使用したスタティック *VTI*



SVTI の基本設定は、仮想ファイアウォール定義を含むように変更されています。

C8000 ルータ設定

```
hostname c8000
.
.
ip inspect max-incomplete high 1000000
ip inspect max-incomplete low 800000
ip inspect one-minute high 1000000
ip inspect one-minute low 800000
ip inspect tcp synwait-time 60
ip inspect tcp max-incomplete host 100000 block-time 2
ip inspect name IOSFW1 tcp timeout 300
ip inspect name IOSFW1 udp
!
.
.
interface GigabitEthernet0/1
 description Internet Connection
 ip address 172.18.143.246 255.255.255.0
 ip access-group 100 in
 ip nat outside
!
interface Tunnel0
 ip address 10.0.51.217 255.255.255.0
 ip nat inside
 ip inspect IOSFW1 in
 tunnel source 10.0.149.217
 tunnel destination 10.0.149.203
 tunnel mode ipsec ipv4
 tunnel protection ipsec profile P1
!
ip classless
ip route 0.0.0.0 0.0.0.0 172.18.143.1
!
ip nat translation timeout 120
ip nat translation finrst-timeout 2
ip nat translation max-entries 300000
ip nat pool test1 10.2.100.1 10.2.100.50 netmask 255.255.255.0
ip nat inside source list 110 pool test1 vrf test-vti1 overload
!
access-list 100 permit esp any any
```

```
access-list 100 permit udp any eq isakmp any
access-list 100 permit udp any eq non500-isakmp any
access-list 100 permit icmp any any
access-list 110 deny   esp any any
access-list 110 deny   udp any eq isakmp any
access-list 110 permit ip any any
access-list 110 deny   udp any eq non500-isakmp any
!
end
```

例：ダイナミック仮想トンネル インターフェイス Easy VPN サーバ

次に、DVTI Easy VPN サーバを使用する例を示します。このサーバは、IPsec リモート アクセス アグリゲータになります。クライアントは、Cisco VPN Client を実行しているホーム ユーザにすることも、Easy VPN クライアントとして設定された Cisco IOS ルータにすることもできます。

C8000 ルータ設定

```
hostname c8000
!
aaa new-model
aaa authentication login local_list local
aaa authorization network local_list local
aaa session-id common
!
ip subnet-zero
ip cef
!
username cisco password 0 cisco123
!
controller ISA 1/1
!
crypto isakmp policy 1
 encr aes
 authentication pre-share
 group 14
!
crypto isakmp client configuration group group1
 key cisco123
 pool group1pool
 save-password
!
crypto isakmp profile vpn1-ra
   match identity group group1
   client authentication list local_list
   isakmp authorization list local_list
   client configuration address respond
   virtual-template 1
!
crypto ipsec transform-set VTI-TS esp-aes esp-sha-hmac
!
crypto ipsec profile test-vti1
 set transform-set VTI-TS
!
interface GigabitEthernet0/1
 description Internet Connection
 ip address 172.18.143.246 255.255.255.0
!
```

```
interface GigabitEthernet0/2
 description Internal Network
 ip address 10.2.1.1 255.255.255.0
!
interface Virtual-Template1 type tunnel
 ip unnumbered GigabitEthernet0/1
 ip virtual-reassembly
 tunnel mode ipsec ipv4
 tunnel protection ipsec profile test-vti1
!
ip local pool group1pool 192.168.1.1 192.168.1.4
ip classless
ip route 0.0.0.0 0.0.0.0 172.18.143.1
!
end
```

例：ダイナミック仮想トンネル インターフェイス Easy VPN サーバの結果の確認

次に、DVTI が、Easy VPN サーバ用に設定されている例を示します。

```
Router# show running-config interface Virtual-Access2

Building configuration...
Current configuration : 250 bytes
!
interface Virtual-Access2
 ip unnumbered GigabitEthernet0/1
 ip virtual-reassembly
 tunnel source 172.18.143.246
 tunnel destination 172.18.143.208
 tunnel mode ipsec ipv4
 tunnel protection ipsec profile test-vti1
 no tunnel protection ipsec initiate
end
Router# show ip route

Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route
Gateway of last resort is 10.2.1.10 to network 0.0.0.0
     172.18.0.0/24 is subnetted, 1 subnets
C       172.18.143.0 is directly connected, GigabitEthernet0/1
     192.168.1.0/32 is subnetted, 1 subnets
S       192.168.1.1 [1/0] via 0.0.0.0, Virtual-Access2
     10.0.0.0/24 is subnetted, 1 subnets
C       10.2.1.0 is directly connected, GigabitEthernet0/2
S*   0.0.0.0/0 [1/0] via 172.18.143.1
```

例：VRF が仮想テンプレートに基づいて設定された場合のダイナミック VTI を使用した VRF 認識 IPsec

次に、仮想テンプレートに基づいて DVTI を利用するように VRF 認識 IPsec を設定する例を示します。

```
hostname c8000
!
ip vrf VRF-100-1
  rd 1:1
!
ip vrf VRF-100-2
  rd 1:1
!
!
!
crypto keyring cisco-100-1
  pre-shared-key address 10.1.1.1 key cisco-100-1
crypto keyring cisco-100-2
  pre-shared-key address 10.1.2.1 key cisco-100-2
crypto isakmp profile cisco-isakmp-profile-100-1
  keyring cisco-100-1
  match identity address 10.1.1.0 255.255.255.0
  virtual-template 101
crypto isakmp profile cisco-isakmp-profile-100-2
  keyring cisco-100-2
  match identity address 10.1.2.0 255.255.255.0
  virtual-template 102
!
!
crypto ipsec transform-set cisco esp-aes esp-sha-hmac
!
crypto ipsec profile cisco-ipsec-profile-101
  set security-policy limit 3
  set transform-set cisco
!
crypto ipsec profile cisco-ipsec-profile-102
  set security-policy limit 5
  set transform-set Cisco
!
interface Virtual-Template101 type tunnel
  ip vrf forwarding VRF-100-1
  ip unnumbered Ethernet 0/0
  tunnel mode ipsec ipv4
  tunnel protection ipsec profile cisco-ipsec-profile-101
!
interface Virtual-Template102 type tunnel
  ip vrf forwarding VRF-100-2
  ip unnumbered Ethernet 0/0
  tunnel mode ipsec ipv4
  tunnel protection ipsec profile cisco-ipsec-profile-102
!
```

例：VRF が仮想テンプレートと IPsec プロファイル内のゲートウェイ オプションに基づいて設定された場合のダイナミック VTI を使用した VRF 認識 IPsec

次に、VRF が仮想テンプレートと IPsec プロファイル内のゲートウェイ オプションに基づいて設定されている場合に、DVTI を利用するように VRF 認識 IPsec を設定する例を示します。

```
hostname c8000
!
ip vrf VRF-100-1
 rd 1:1
!
ip vrf VRF-100-2
 rd 1:1
!
!
!
crypto keyring cisco-100-1
 pre-shared-key address 10.1.1.1 key cisco-100-1
crypto keyring cisco-100-2
 pre-shared-key address 10.1.2.1 key cisco-100-2
crypto isakmp profile cisco-isakmp-profile-100-1
 keyring cisco-100-1
 match identity address 10.1.1.0 255.255.255.0
 virtual-template 101
crypto isakmp profile cisco-isakmp-profile-100-2
 keyring cisco-100-2
 match identity address 10.1.2.0 255.255.255.0
 virtual-template 102
!
!
crypto ipsec transform-set cisco esp-3des esp-sha-hmac
!
crypto ipsec profile cisco-ipsec-profile-101
 set security-policy limit 3
 set transform-set cisco
 set reverse-route gateway 172.16.0.1
!
crypto ipsec profile cisco-ipsec-profile-102
 set security-policy limit 5
 set transform-set cisco
 set reverse-route gateway 172.16.0.1
!
interface Virtual-Template101 type tunnel
 ip vrf forwarding VRF-100-1
 ip unnumbered Ethernet 0/0
 tunnel mode ipsec ipv4
 tunnel protection ipsec profile cisco-ipsec-profile-101
!
interface Virtual-Template102 type tunnel
 ip vrf forwarding VRF-100-2
 ip unnumbered Ethernet 0/0
 tunnel mode ipsec ipv4
 tunnel protection ipsec profile cisco-ipsec-profile-102
!
```

例：VRF が ISAKMP プロファイルに基づいて設定された場合のダイナミック VTI を使用した VRF 認識 IPsec

```
hostname c8000
!
ip vrf VRF-100-1
  rd 1:1
!
ip vrf VRF-100-2
  rd 1:1
```

```
!
crypto keyring cisco-100-1
  pre-shared-key address 10.1.1.1 key cisco-100-1
crypto keyring cisco-100-2
  pre-shared-key address 10.1.2.1 key cisco-100-2
crypto isakmp profile cisco-isakmp-profile-100-1
  vrf VRF-100-1
  keyring cisco-100-1
  match identity address 10.1.1.0 255.255.255.0
  virtual-template 1
crypto isakmp profile cisco-isakmp-profile-100-2
  vrf VRF-100-2
  keyring cisco-100-2
  match identity address 10.1.2.0 255.255.255.0
  virtual-template 1
!
!
crypto ipsec transform-set cisco esp-aes esp-sha-hmac
crypto ipsec profile cisco-ipsec-profile
  set security-policy limit 3
  set transform-set cisco
!
!
!
interface Virtual-Template 1 type tunnel
  ip unnumbered ethernet 0/0
  tunnel mode ipsec ipv4
  tunnel protection ipsec profile cisco-ipsec-profile
!
!
```

例：VRF が ISAKMP プロファイルと IPsec プロファイル内のゲートウェイ オプションに基づいて設定された場合のダイナミック VTI を使用した VRF 認識 IPsec

次に、VRF が ISAKMP プロファイルと IPsec プロファイル内のゲートウェイ オプションに基づいて設定されている場合に、DVTI を利用するように VRF 認識 IPsec を設定する例を示します。

```
hostname C8000 server
!
ip vrf VRF-100-1
 rd 1:1
!
ip vrf VRF-100-2
 rd 1:1
!
crypto keyring cisco-100-1
 pre-shared-key address 10.1.1.1 key cisco-100-1
crypto keyring cisco-100-2
 pre-shared-key address 10.1.2.1 key cisco-100-2
crypto isakmp profile cisco-isakmp-profile-100-1
 vrf VRF-100-1
 keyring cisco-100-1
 match identity address 10.1.1.0 255.255.255.0
```

```
 virtual-template 1
crypto isakmp profile cisco-isakmp-profile-100-2
 vrf VRF-100-2
 keyring cisco-100-2
 match identity address 10.1.2.0 255.255.255.0
 virtual-template 1
!
!
crypto ipsec transform-set cisco esp-3des esp-sha-hmac
crypto ipsec profile cisco-ipsec-profile
 set security-policy limit 3
 set transform-set cisco
 set reverse-route gateway 172.16.0.1
!
!
!
interface Virtual-Template1 type tunnel
 ip unnumbered Ethernet 0/0
 tunnel mode ipsec ipv4
 tunnel protection ipsec profile cisco-ipsec-profile
!
!
```

例：VRF が仮想テンプレートと ISAKMP プロファイルの両方に基づいて設定された場合のダイナミック VTI を使用した VRF 認識 IPsec

（注） ISAKMP プロファイルと仮想テンプレートに基づいて別々の VRF が設定されている場合は、仮想テンプレートに基づいて設定された VRF が優先されます。この設定は推奨されません。

次に、VRF が仮想テンプレートと ISAKMP プロファイルの両方に基づいて設定されている場合に、DVTI を利用するように VRF 認識 IPsec を設定する例を示します。

```
hostname C8000 server
.
.
.
ip vrf test-vti2
 rd 1:2
 route-target export 1:1
 route-target import 1:1
!
.
.
.
ip vrf test-vti1
 rd 1:1
 route-target export 1:1
 route-target import 1:1
!
.
.
.
crypto isakmp profile cisco-isakmp-profile
```

```
 vrf test-vti2
 keyring key
 match identity address 10.1.1.0 255.255.255.0
!
.
.
.
interface Virtual-Template1 type tunnel
 ip vrf forwarding test-vti1
 ip unnumbered Loopback 0
 ip virtual-reassembly
 tunnel mode ipsec ipv4
 tunnel protection ipsec profile test-vti1
!
.
.
.
end
```

例：仮想ファイアウォールを使用したダイナミック仮想トンネルインターフェイス

DVTI Easy VPN サーバは、仮想ファイアウォールの背後に設定できます。Behind-the-firewall 設定を使用すれば、ユーザはネットワークに入れますが、ネットワーク ファイアウォールは不正アクセスから保護されます。仮想ファイアウォールでは、コンテキスト ベースのアクセス コントロール（CBAC）と、インターネットインターフェイスおよび仮想テンプレートに対して適用される NAT が使用されます。

```
hostname c8000
.
.
ip inspect max-incomplete high 1000000
ip inspect max-incomplete low 800000
ip inspect one-minute high 1000000
ip inspect one-minute low 800000
ip inspect tcp synwait-time 60
ip inspect tcp max-incomplete host 100000 block-time 2
ip inspect name IOSFW1 tcp timeout 300
ip inspect name IOSFW1 udp
!
.
.
interface GigabitEthernet0/1
 description Internet Connection
 ip address 172.18.143.246 255.255.255.0
 ip access-group 100 in
 ip nat outside
!
interface GigabitEthernet0/2
 description Internal Network
 ip address 10.2.1.1 255.255.255.0
!
interface Virtual-Template1 type tunnel
 ip unnumbered Loopback0
 ip nat inside
 ip inspect IOSFW1 in
 tunnel mode ipsec ipv4
 tunnel protection ipsec profile test-vti1
```

```
!
ip classless
ip route 0.0.0.0 0.0.0.0 172.18.143.1
!
ip nat translation timeout 120
ip nat translation finrst-timeout 2
ip nat translation max-entries 300000
ip nat pool test1 10.2.100.1 10.2.100.50 netmask 255.255.255.0
ip nat inside source list 110 pool test1 vrf test-vti1 overload
!
access-list 100 permit esp any any
access-list 100 permit udp any eq isakmp any
access-list 100 permit udp any eq non500-isakmp any
access-list 100 permit icmp any any
access-list 110 deny   esp any any
access-list 110 deny   udp any eq isakmp any
access-list 110 permit ip any any
access-list 110 deny   udp any eq non500-isakmp any
!
end
```

例：QoS を使用したダイナミック仮想トンネル インターフェイス

サービス ポリシーを仮想テンプレートに適用することによって、QoS を DVTI トンネルに追加できます。テンプレートを複製して仮想アクセスインターフェイスを作成した場合は、サービス ポリシーが仮想アクセス インターフェイスにも適用されます。次に、QoS が追加された DVTI 基本設定の例を示します。

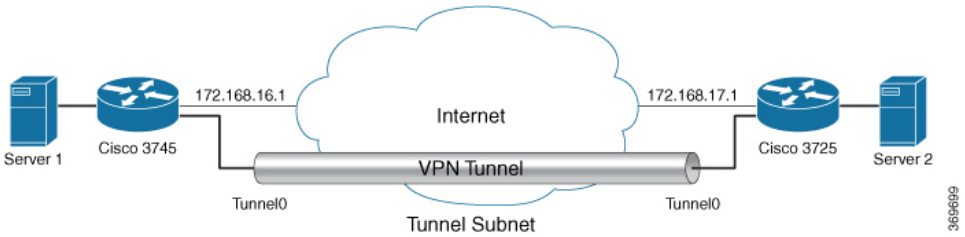
```
hostname c8000
.
.
class-map match-all VTI
 match any
!
policy-map VTI
  class VTI
  police cir 2000000
    conform-action transmit
    exceed-action drop
!
.
.
interface Virtual-Template1 type tunnel
 ip vrf forwarding test-vti1
 ip unnumbered Loopback0
 ip virtual-reassembly
 tunnel mode ipsec ipv4
 tunnel protection ipsec profile test-vti1
 service-policy output VTI
!
.
.
!
end
```

例：複数の IPsec SA を使用したスタティック仮想トンネルインターフェイス

次の例では、SVTI を使用して Cisco 3745 と Cisco 3725 の 2 つのルータの間で IPSec トンネルを確立します。この設定では、非 any-any トラフィックセレクタを使用し、複数の IPSec SA の形成を有効にします。

IPv4 トンネルモードのルータでの設定例：

次の図は、設定の参照トポロジを示しています。



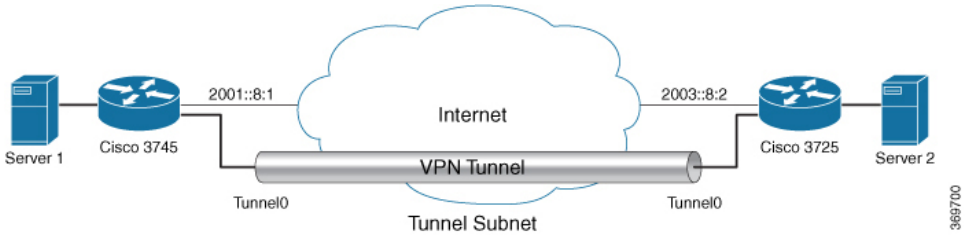
Cisco 3745 ルータの設定例は、次のとおりです。

```
crypto isakmp policy 1
 authentication pre-share
 group 2
!
crypto isakmp policy 5
 encr 3des
 authentication pre-share
 group 2
crypto isakmp key example address 172.168.17.1
!
!
crypto ipsec transform-set svti1 esp-3des esp-sha-hmac
 mode tunnel
!
!
crypto ipsec profile ipsec_prof
 set transform-set svti1
!
!
!
interface Loopback0
 ip address 30.0.0.1 255.255.255.0
!
interface Loopback1
 ip address 50.0.0.1 255.255.255.0
!
interface Tunnel0
 ip address 11.1.1.2 255.255.255.0
 tunnel source Ethernet0/0
 tunnel mode ipsec ipv4
 tunnel destination 172.168.17.1
 tunnel protection ipsec policy ipv4 ipsec_acl1
 tunnel protection ipsec profile ipsec_prof
!
interface Ethernet0/0
 ip address 172.168.16.1 255.255.255.0
!
```

```
!
ip access-list extended ipsec_acl1
permit ip 30.0.0.0 0.0.0.255 40.0.0.0 0.0.0.255
permit ip 50.0.0.0 0.0.0.255 60.0.0.0 0.0.0.255
```

IPv6 トンネルモードのルータでの設定例：

次の図は、設定の参照トポロジを示しています。



Cisco 3745 ルータの設定例は、次のとおりです。

```
crypto isakmp policy 1
 authentication pre-share
 group 2
!
crypto isakmp policy 5
 encr 3des
 authentication pre-share
 group 2
crypto isakmp key example address ipv6 2003::8:2/112
!
!
crypto ipsec transform-set svti1 esp-3des esp-sha-hmac
 mode tunnel
!
!
crypto ipsec profile ipsec_prof
 set transform-set svti1
!
!
!
interface Loopback0
 ipv6 address 2005::10:1/112
 ipv6 enable
!
interface Loopback1
 ipv6 address 2005::15:1/112
 ipv6 enable
!
interface Loopback2
 ipv6 address 2005::20:1/112
 ipv6 enable
!
interface Tunnel0
 ip address 11.1.1.2 255.255.255.0
 ipv6 address 400::10:1/112
 ipv6 enable
 tunnel source Ethernet0/0
 tunnel mode ipsec ipv6
 tunnel destination 2003::8:2
 tunnel protection ipsec policy ipv6 ipsec_acl2
 tunnel protection ipsec profile ipsec_prof
!
```

```
interface Ethernet0/0
 ipv6 address 2001::8:1/112
 ipv6 enable
!
!
ipv6 access-list ipsec_acl2
sequence 10 permit ipv6 host 2005::10:1 host 2005::11:1
sequence 20 permit ipv6 host 2005::15:1 host 2005::16:1
sequence 30 permit ipv6 host 2005::20:1 host 2005::21:1
```

例：デュアルオーバーレイとしてのトンネルモードの設定

次に、トンネルモードをデュアルオーバーレイとして設定する例を示します。

```
Device# configure terminal
Router(config)# interface tunnel 1
Router(config-if)# ipv6 enable
Router(config-if)# tunnel source ethernet 0/0
Router(config-if)# tunnel mode ipsec dual-overlay
Router(config-if)# tunnel destination 89.89.89.1 255.255.255.255.0
Device(config-if)# tunnel protection IPsec profile ipsecprof
```

デュアルオーバーレイとしてのトンネルモードの設定の確認

次のコマンドを使用して、設定をトラブルシューティングします。

- **show crypto session [detail]**
- **show crypto ipsec sa**
- **show crypto map**
- **show crypto socket**
- **show crypto ikev2 session [detail]**

```
Device# show crypto map
Crypto Map: "Tunnel0-head-0" IKEv2 profile: prof

Crypto Map IPv4 "Tunnel0-head-0" 65536 ipsec-isakmp
 IKEv2 Profile: prof
 Profile name: prof
 Security association lifetime: 4608000 kilobytes/120 seconds
 Dualstack (Y/N): N

 Responder-Only (Y/N): N
 PFS (Y/N): N
 Mixed-mode : Disabled
 Transform sets={
  default:  { esp-aes esp-sha-hmac  } ,
 }

Crypto Map IPv4 "Tunnel0-head-0" 65537 ipsec-isakmp
 Map is a PROFILE INSTANCE.
 Peer = 10.10.10.2
 IKEv2 Profile: prof
 Extended IP access list
     access-list  permit ip any any
 Current peer: 10.10.10.2
 Security association lifetime: 4608000 kilobytes/120 seconds
       Dualstack (Y/N): Y
```

```
        TRUE  ident (addr/mask/prot/port): {LOCAL -> REMOTE}
              0.0.0.0/0.0.0.0/0/0 -> 0.0.0.0/0.0.0.0/0/0
              ::/0.0.0.0/0/0 -> ::/0/0/0
 Responder-Only (Y/N): N
 PFS (Y/N): N
 Mixed-mode : Disabled
 Transform sets={
  default:  { esp-aes esp-sha-hmac  } ,
 }
 Always create SAs
 Interfaces using crypto map Tunnel0-head-0:
  Tunnel0
```

```
Device# show  crypto ipsec sa

interface: Tunnel0
    Crypto map tag: Tunnel0-head-0, local addr 10.10.10.1

   protected vrf: (none)
   local  ident (addr/mask/prot/port): (0.0.0.0/0.0.0.0/0/0)
   remote ident (addr/mask/prot/port): (0.0.0.0/0.0.0.0/0/0)
    TRUE  ident (addr/mask/prot/port): {LOCAL -> REMOTE}
              0.0.0.0/0.0.0.0/0/0 -> 0.0.0.0/0.0.0.0/0/0
              ::/0.0.0.0/0/0 -> ::/0/0/0
   current_peer 10.10.10.2 port 500
     PERMIT, flags={origin_is_acl,}
    #pkts encaps: 0, #pkts encrypt: 0, #pkts digest: 0
    #pkts decaps: 0, #pkts decrypt: 0, #pkts verify: 0
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0
    #pkts not decompressed: 0, #pkts decompress failed: 0
    #send errors 0, #recv errors 0

     local crypto endpt.: 10.10.10.1, remote crypto endpt.: 10.10.10.2
     plaintext mtu 1438, path mtu 1500, ip mtu 1500, ip mtu idb Ethernet0/0
     current outbound spi: 0x4776A36B(1198957419)
     PFS (Y/N): N, DH group: none

     inbound esp sas:
      spi: 0xA97EDEE7(2843664103)
        transform: esp-aes esp-sha-hmac ,
        in use settings ={Tunnel, }
       conn id: 4, flow_id: 4, sibling_flags FFFFFFFF80000040, crypto map: Tunnel0-head-0

         sa timing: remaining key lifetime (k/sec): (4377587/76)
        IV size: 16 bytes
        replay detection support: Y
        Status: ACTIVE(ACTIVE)

     inbound ah sas:

     inbound pcp sas:

     outbound esp sas:
      spi: 0x4776A36B(1198957419)
        transform: esp-aes esp-sha-hmac ,
        in use settings ={Tunnel, }
       conn id: 3, flow_id: 3, sibling_flags FFFFFFFF80000040, crypto map: Tunnel0-head-0

         sa timing: remaining key lifetime (k/sec): (4377587/76)
        IV size: 16 bytes
        replay detection support: Y
        Status: ACTIVE(ACTIVE)
```

```
     outbound ah sas:

     outbound pcp sas:
Device# show crypto socket

Number of Crypto Socket connections 1

   Tu0 Peers (local/remote): 10.10.10.1/10.10.10.2
       Local Ident  (addr/mask/port/prot): (0.0.0.0/0.0.0.0/0/0)
       Remote Ident (addr/mask/port/prot): (0.0.0.0/0.0.0.0/0/0)
        TRUE  ident (addr/mask/prot/port): {LOCAL -> REMOTE}
             0.0.0.0/0.0.0.0/0/0 -> 0.0.0.0/0.0.0.0/0/0
             ::/0.0.0.0/0/0 -> ::/0/0/0
       IPSec Profile: "prof"
       Socket State: Open
       Client: "TUNNEL SEC" (Client State: Active)
Crypto Sockets in Listen state:
Client: "TUNNEL SEC" Profile: "prof" Map-name: "Tunnel0-head-0"

Device# show cry ikev2 session
 IPv4 Crypto IKEv2 Session

Session-id:1, Status:UP-ACTIVE, IKE count:1, CHILD count:1

Tunnel-id Local                 Remote                fvrf/ivrf            Status
1         10.10.10.1/500        10.10.10.2/500        none/none            READY
      Encr: AES-CBC, keysize: 256, PRF: SHA512, Hash: SHA512, DH Grp:19, Auth sign: PSK,
 Auth verify: PSK
      Life/Active Time: 86400/145 sec
      CE id: 1001, Session-id: 1
      Local spi: 25A0B173944015D3       Remote spi: 9F0C7677425670E1
Child sa:
          local selector  0.0.0.0/0 - 255.255.255.255/65535
          local selector  ::/0 - FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF/65535
          remote selector 0.0.0.0/0 - 255.255.255.255/65535
          remote selector ::/0 - FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF/65535
          ESP spi in/out: 0xA97EDEE7/0x4776A36B

 IPv6 Crypto IKEv2 Session

Device# show crypto session
Crypto session current status

Interface: Tunnel0
Profile: prof
Session status: UP-ACTIVE
Peer: 10.10.10.2 port 500
  Session ID: 1
  IKEv2 SA: local 10.10.10.1/500 remote 10.10.10.2/500 Active
  IPSEC FLOW: permit ip   0.0.0.0/0.0.0.0 0.0.0.0/0.0.0.0
  TRUE IDENT (addr/mask/prot/port): {LOCAL -> REMOTE}
             0.0.0.0/0.0.0.0/0/0 -> 0.0.0.0/0.0.0.0/0/0
             ::/0.0.0.0/0/0 -> ::/0/0/0
        Active SAs: 2, origin: crypto map
```

IPsec 仮想トンネル インターフェイスに関する追加のリファレンス

関連資料

関連項目	マニュアル タイトル
Cisco IOS コマンド	『Cisco IOS Master Command List, All Releases』
セキュリティ コマンド	• 『Cisco IOS Security Command Reference Commands A to C』 • 『Cisco IOS Security Command Reference Commands D to L』 • 『Cisco IOS Security Command Reference Commands M to R』 • 『Cisco IOS Security Command Reference Commands S to Z』
IPsec の設定	『*Configuring Security for VPNs with IPsec*』
QoS の設定	『*Cisco IOS Quality of Service Solutions Configuration Guide*』
EasyVPN の設定	• 「*Cisco Easy VPN Remote*」 • 「*Easy VPN Server*」
推奨される暗号化アルゴリズム	『Next Generation Encryption』

標準および RFC

標準/RFC	タイトル
RFC 2401	『*Security Architecture for the Internet Protocol*』
RFC 2408	『*Internet Security Association and Key Management Protocol*』
RFC 2409	『*The Internet Key Exchange (IKE)*』

シスコのテクニカル サポート

説明	リンク
右の URL にアクセスして、シスコのテクニカル サポートを最大限に活用してください。これらのリソースは、ソフトウェアをインストールして設定したり、シスコの製品やテクノロジーに関する技術的問題を解決したりするために使用してください。この Web サイト上のツールにアクセスする際は、Cisco.com のログイン ID およびパスワードが必要です。	http://www.cisco.com/cisco/web/support/index.html

IPsec 仮想トンネル インターフェイスに関する機能情報

次の表に、このモジュールで説明した機能に関するリリース情報を示します。この表は、ソフトウェア リリース トレインで各機能のサポートが導入されたときのソフトウェア リリースだけを示しています。その機能は、特に断りがない限り、それ以降の一連のソフトウェア リリースでもサポートされます。

プラットフォームのサポートおよびシスコ ソフトウェアイメージのサポートに関する情報を検索するには、Cisco Feature Navigator を使用します。Cisco Feature Navigator にアクセスするには、www.cisco.com/go/cfn に移動します。Cisco.com のアカウントは必要ありません。

表 *1 : IPsec* 仮想トンネル インターフェイスに関する機能情報

機能名	リリース	機能の設定情報
ダイナミック IPsec VTI	12.3(7)T 12.3(14)T	ダイナミック VTI によって IP アドレスの使用が効率的になり、セキュアな接続が提供されます。ダイナミック VTI によって、動的にダウンロード可能な、グループごとおよびユーザごとのポリシーを RADIUS サーバ上で設定できます。IPsec ダイナミック VTI を使用すれば、リモート アクセス VPN 用の高度にセキュアな接続を構築することができます。ダイナミック VTI によって、VRF 認識 IPsec の導入が簡単になります。 次のコマンドが導入または変更されました。 **crypto isakmp profile, interface virtual-template, show vtemplate, tunnel mode, virtual-template.**
FlexVPN 混合モードのサポート	15.4(2)T Cisco IOS XE Release 3.10S	FlexVPN 混合モード機能は、IPsec IPv6 トランスポート経由の IPv4 トラフィックの伝送をサポートします。これは、IPsec スタック上でのデュアル スタックのサポートにつながる第 1 段階です。この実装は、IPv4 トラフィックと IPv6 トラフィックの両方に対する単一の IPSec セキュリティ アソシエーション（SA）ペアの使用をサポートしません。 この機能は、IKEv2 とダイナミック VTI を使用したリモート アクセス VPN に対してのみサポートされます。

機能名	リリース	機能の設定情報
ダイナミック VTI に対するマルチ SA	15.2(1)T Cisco IOS XE Release 3.2S	DVTIは、イニシエータから提案された複数のIPsecセレクタを受け入れることができます。 次のコマンドが導入または変更されました。 **set security-policy limit, set reverse-route.**
スタティック IPsec VTI	12.2(33)SRA 12.2(33)SXH 12.3(7)T 12.3(14)T Cisco IOS XE Release 2.1	IPsec VTI では、IPsec トンネルを終端するためのルーティング可能なインターフェイス タイプと、オーバーレイ ネットワークを形成するためにサイト間の保護を定義する簡単な手段が提供されます。IPsec VTI によって、リモート リンクを保護するためのIPsecの設定が簡素化され、マルチキャストがサポートされ、さらには、ネットワーク管理およびロード バランシングが簡単に実現できるようになります。
トンネル モード自動選択	15.4(2)T Cisco IOS XE リリース 3.12S	トンネル モード自動選択機能は、設定を容易にし、レスポンダの詳細の入手を支援します。この機能は、IKEプロファイルから仮想アクセス インターフェイスが作成されるとすぐに、トンネリング プロトコル（GREまたはIPsec）とトランスポート プロトコル（IPv4 または IPv6）を自動的に仮想テンプレートに適用します。 次のコマンドが導入または変更されました： **virtual-template**

機能名	リリース	機能の設定情報
FlexVPN 混合モード v4 経由 v6 トランスポート	Cisco IOS XE Everest 16.4.1	FlexVPN 混合モード v4 経由 v6 トランスポート機能は、IPsec IPv4 トランスポート経由の IPv6 トラフィックの伝送をサポートします。この実装は、IPv4 トラフィックと IPv6 トラフィックの両方に対する単一の IPSec セキュリティ アソシエーション（SA）ペアの使用をサポートしません。
Cisco 以外のデバイスでの IPsec デュアルスタックのサポート	Cisco IOS XE Cupertino 17.9.x	この機能により、IPv4 を介してトンネリングされる単一の IPsec セキュリティ アソシエーション（SA）を使用して IPv4 トラフィックと IPv6 トラフィックの両方を伝送することが可能になります。IOS XE リリース 17.9.1a 以降、シスコでは、トンネルインターフェイスの入力側がサードパーティの IPsec クライアントで設定されている場合、アクセス制御リストの特定のサブネットをサポートしています。SVTI シングル セキュリティ アソシエーションデュアル スタック機能の導入により、Business-to-Business（B2B）サービスやその他の IoT ビジネスを効率的に管理できるようになりました。

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。