Cisco Catalyst IE3x00 高耐久性、IE3400 Heavy Duty、ESS3300 シリーズスイッチ セキュリティ コンフィギュレーション ガイド

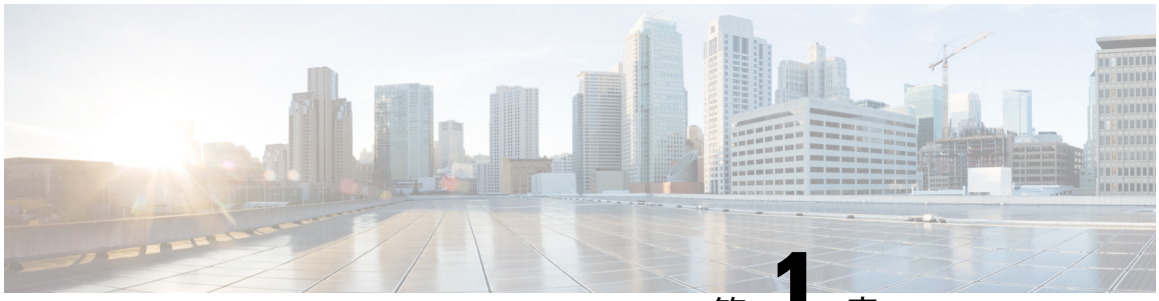
初版：2020 年 8 月 10 日

最終更新：2021 年 8 月 2 日

シスコシステムズ合同会社
〒107-6227 東京都港区赤坂9-7-1 ミッドタウン・タワー
http://www.cisco.com/jp
お問い合わせ先：シスコ コンタクトセンター
0120-092-255 （フリーコール、携帯・PHS含む）
電話受付時間：平日 10:00～12:00、13:00～17:00
http://www.cisco.com/jp/go/contactcenter/

【注意】シスコ製品をご使用になる前に、安全上の注意（**www.cisco.com/jp/go/safety_warning/**）をご確認ください。本書は、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。また、契約等の記述については、弊社販売パートナー、または、弊社担当者にご確認ください。

注：この製品のマニュアルセットは、偏向のない言語を使用するように配慮されています。このドキュメントセットでの偏向のない言語とは、年齢、障害、性別、人種的アイデンティティ、民族的アイデンティティ、性的指向、社会経済的地位、およびインターセクショナリティに基づく差別を意味しない言語として定義されています。製品ソフトウェアのユーザインターフェイスにハードコードされている言語、RFPのドキュメントに基づいて使用されている言語、または参照されているサードパーティ製品で使用されている言語によりドキュメントに例外が存在する場合があります。



第 1 章

IPv6 ファースト ホップ セキュリティの設定

IPv6 でのファースト ホップ セキュリティの前提条件

必要な、IPv6 が有効になっている SDM テンプレートが設定されていること。

IPv6 でのファースト ホップ セキュリティの制約事項

- 次の制限は、FHS ポリシーを EtherChannel インターフェイスに適用する場合に該当します（ポート チャネル）。
  - FHS ポリシーがアタッチされた物理ポートは EtherChannel グループに参加することができません。

- FHS ポリシーは、EtherChannel グループのメンバーである場合に物理ポートにアタッチすることができません。

- デフォルトでは、スヌーピング ポリシーにはセキュリティ レベルのガードがあります。そのようなスヌーピング ポリシーがアクセス スイッチに設定されると、ルータまたは DHCP サーバ/リレーに対応するアップリンク ポートが信頼できるポートとして設定されていても、IPv6（DHCPv6）サーバ パケットに対する外部 IPv6 ルータ アドバタイズメント(RA)または Dynamic Host Configuration Protocol はブロックされます。IPv6 RA または DHCPv6 サーバ メッセージを許可するには、次の手順を実行します。
  - IPv6 RA ガード ポリシー（RA の場合）または IPv6 DHCP ガード ポリシー（DHCP サーバ メッセージの場合）をアップリンク ポートに適用します。
  - 低いセキュリティ レベルでスヌーピング ポリシーを設定します（たとえば、glean や inspect など）。しかし、ファースト ホップ セキュリティ機能の利点が有効でないため、このようなスヌーピング ポリシーでは、低いセキュリティ レベルを設定することはお勧めしません。

- 同じノードにおけるホストとガードの設定はサポートされていません。
- DHCPv6 ガードを機能させるには、同じスイッチ上の対応する VLAN で SVI を設定する必要があります。

IPv6 でのファースト ホップ セキュリティに関する情報

IPv6 のファーストホップ セキュリティ（FHS IPv6）は、ポリシーを物理インターフェイス、または VLAN にアタッチできる一連の IPv6 セキュリティ機能です。IPv6 ソフトウェア ポリシー データベース サービスは、これらのポリシーを保存しアクセスします。ポリシーを設定または変更すると、ポリシー属性はソフトウェア ポリシー データベースに保存または更新され、その後指定したとおりに適用されます。次の IPv6 ポリシーが現在サポートされています。

- IPv6 スヌーピング ポリシー：IPv6 スヌーピング ポリシーは、IPv6 内の FHS で使用できるほとんどの機能を有効にできるコンテナ ポリシーとして機能します。

（注） 以降、IPv6 スヌーピングポリシー機能は廃止され、Switch Integrated Security Features（SISF）ベースのデバイス追跡に置き換わります。IPv6 スヌーピング ポリシー コマンドは CLI で引き続き使用でき、既存の設定は引き続きサポートされますが、コマンドは今後のリリースで CLI から削除されます。代わりの機能の詳細については、このガイドの「*SISF* ベースのデバイス追跡の設定」を参照してください。

- IPv6 FHS バインディング テーブルの内容：スイッチに接続された IPv6 ネイバーのデータベース テーブルはネイバー探索（ND）プロトコル スヌーピングなどの情報ソースから作

成されます。このデータベースまたはバインディング テーブルは、リンク層アドレス（LLA）、IPv4 または IPv6 アドレス、およびスプーフィングやリダイレクト攻撃を防止するためにネイバーのプレフィックス バインディングを検証するために、さまざまな IPv6 ガード機能（IPv6 ND 検査など）によって使用されます。

（注） 以降、IPv6 FHS バインディング テーブル コンテンツ機能は、SISF ベースのデバイス追跡によってサポートされます。詳細については、このガイドの「*SISF* ベースのデバイス追跡の設定」を参照してください。

- IPv6 ネイバー探索検査：IPv6 ND 検査は、レイヤ 2 ネイバー テーブル内のステートレス自動設定アドレスのバインディングを学習し、保護します。IPv6 ND 検査は、信頼できるバインディング テーブル データベースを構築するためにネイバー探索メッセージを分析します。準拠していない IPv6 ネイバー探索メッセージは破棄されます。ND メッセージは、その IPv6 からメディア アクセス コントロール（MAC）へのマッピングが検証可能な場合に信頼できると見なされます。

  この機能によって、DAD、アドレス解決、ルータ ディスカバリ、ネイバー キャッシュに対する攻撃などの、ND メカニズムに固有の脆弱性のいくつかが軽減されます。

（注） Cisco IOS XE Amsterdam 17.1.1 以降、IPv6 ND 検査機能は廃止され、SISF ベースのデバイス追跡機能に置き換えられます。IPv6 ND 検査コマンドは CLI で引き続き使用でき、既存の設定は引き続きサポートされますが、コマンドは今後のリリースで CLI から削除されます。代わりの機能の詳細については、このガイドの「*SISF* ベースのデバイス追跡の設定」を参照してください。

- IPv6 ルータ アドバタイズメント ガード：IPv6 ルータ アドバタイズメント（RA）ガード機能を使用すると、ネットワーク管理者は、ネットワーク スイッチ プラットフォームに到着した不要または不正な RA ガード メッセージをブロックまたは拒否できます。RA は、リンクで自身をアナウンスするためにルータによって使用されます。RA ガード機能は、これらの RA を分析して、未承認のルータによって送信された偽の RA をフィルタリングして除外します。ホスト モードでは、ポートではルータ アドバタイズメントとルータ リダイレクト メッセージはすべて許可されません。RA ガード機能は、レイヤ 2 デバイスの設定情報を、受信した RA フレームで検出された情報と比較します。レイヤ 2 デバイスは、RA フレームとルータ リダイレクト フレームの内容を設定と照らし合わせて検証した後で、RA をユニキャストまたはマルチキャストの宛先に転送します。RA フレームの内容が検証されない場合は、RA は破棄されます。

- IPv6 DHCP ガード：IPv6 DHCP ガード機能は、承認されない DHCPv6 サーバおよびリレー エージェントからの返信およびアドバタイズメント メッセージをブロックします。IPv6 DHCP ガードは、偽造されたメッセージがバインディング テーブルに入るのを防ぎ、DHCPv6 サーバまたは DHCP リレーからデータを受信することが明示的に設定されていな

いポートで受信された DHCPv6 サーバ メッセージをブロックできます。この機能を使用するには、ポリシーを設定してインターフェイスまたは VLAN にアタッチします。DHCP ガード パケットをデバッグするには、**debug ipv6 snooping dhcp-guard** 特権 EXEC コマンドを使用します。

IPv6 スヌーピング ポリシーの設定方法

IPv6 スヌーピングポリシー機能は廃止されました。コマンドは CLI に表示され、設定できますが、代わりにスイッチ統合セキュリティ機能（SISF）ベースのデバイス追跡機能を使用することを推奨します。

IPv6 スヌーピング ポリシーを設定するには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **configure terminal**
2. **ipv6 snooping policy** *policy-name*
3. {[**default**] | [**device-role** {**node** | **switch**}] | [**limit address-count** *value*] | [**no**] | [**protocol** {**dhcp** | **ndp**}] | [**security-level** {**glean** | **guard** | **inspect**}] | [**tracking** {**disable** [**stale-lifetime** [*seconds* | **infinite**] | **enable** [**reachable-lifetime** [*seconds* | **infinite**] }] | [**trusted-port**] }
4. **end**
5. **show ipv6 snooping policy** *policy-name*

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 terminal** 例： `# 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	**ipv6 snooping policy** *policy-name* 例： `(config)# ipv6 snooping policy example_policy`	スヌーピング ポリシーを作成し、IPv6 スヌーピング ポリシー コンフィギュレーション モードに移行します。
ステップ 3	{[**default**]	[**device-role** {**node**	**switch**}]	[**limit address-count** *value*]	[**no**]	[**protocol** {**dhcp**	**ndp**}]	[**security-level** {**glean**	**guard**	**inspect**}]	[**tracking** {**disable** [**stale-lifetime** [*seconds*	**infinite**]	**enable** [**reachable-lifetime** [*seconds*	**infinite**] }]	[**trusted-port**] } 例： `(config-ipv6-snooping)# security-level inspect` 例：	データ アドレス グリーニングを有効にし、さまざまな条件に対してメッセージを検証し、メッセージのセキュリティ レベルを指定します。 • （任意）**default**：すべてをデフォルト オプションに設定します。 • （任意）**device-role**{**node**]	**switch**}：ポートに接続されたデバイスの役割を指定します。デフォルトは **node** です。

	コマンドまたはアクション	目的
	(config-ipv6-snooping)# trusted-port	• （任意）**limit address-count** *value*：ターゲットごとに許可されるアドレス数を制限します。 • （任意）**no**：コマンドを無効にするか、またはそのデフォルトに設定します。 • （任意）**protocol**{**dhcp** | **ndp**}：分析のために、スヌーピング機能にどのプロトコルをリダイレクトするかを指定します。デフォルトは、**dhcp** および **ndp** です。デフォルトを変更するには、**no protocol** コマンドを使用します。 • （任意）**security-level**{**glean**|**guard**|**inspect**}：この機能によって適用されるセキュリティのレベルを指定します。デフォルトは **guard** です。 **glean**：メッセージからアドレスを収集し、何も確認せずにバインディング テーブルに入力します。 **guard**：アドレスを収集し、メッセージを検査します。さらに、RA および DHCP サーバ メッセージを拒否します。これがデフォルトのオプションです。 **inspect**：アドレスを収集し、メッセージの一貫性と準拠を検証して、アドレスの所有権を適用します。 • （任意）**tracking** {**disable** | **enable**}：デフォルトの追跡動作を上書きし、追跡オプションを指定します。 • （任意）**trusted-port**：信頼できるポートを設定します。これにより、該当するターゲットに対するガードが無効になります。信頼できるポートを経由して学習されたバインディングは、他のどのポートを経由して学習されたバインディングよりも優先されます。テーブル内にエントリを作成しているときに衝突が発生した場合、信頼できるポートが優先されます。
ステップ 4	**end** 例： `(config-ipv6-snooping)# exit`	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。
ステップ 5	**show ipv6 snooping policy** *policy-name* 例：	スヌーピング ポリシー設定を表示します。

	コマンドまたはアクション	目的
	#**show ipv6 snooping policy example_policy**	

次のタスク

IPv6 スヌーピング ポリシーをインターフェイスまたは VLAN にアタッチします。

IPv6 スヌーピング ポリシーをインターフェイスにアタッチする方法

インターフェイスまたはVLANにIPv6スヌーピングポリシーをアタッチするには、特権EXECモードで次の手順を実行してください。

手順の概要

1. **configure terminal**
2. **interface** Interface_type *stack/module/port*
3. **switchport**
4. **ipv6 snooping** [**attach-policy** *policy_name* [**vlan** {*vlan_id* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids*}] | **vlan** {*vlan_id* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
5. **do show running-config**

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 terminal** 例： # **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** Interface_type *stack/module/port* 例： (config)# **interface gigabitethernet 1/1/4**	インターフェイスのタイプおよび ID 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 3	**switchport** 例：	switchport モードを開始します。

	コマンドまたはアクション	目的
	(config-if)# **switchport**	（注） インターフェイスがレイヤ3モードの場合に、レイヤ2パラメータを設定するには、パラメータを指定せずに switchport インターフェイス コンフィギュレーション コマンドを入力し、インターフェイスをレイヤ2モードにする必要があります。これにより、インターフェイスがいったんシャットダウンしてから再度有効になり、インターフェイスが接続しているデバイスに関するメッセージが表示されることがあります。レイヤ3モードのインターフェイスをレイヤ2モードにした場合、影響のあるインターフェイスに関連する以前の設定情報が消失する可能性があり、インターフェイスはデフォルト設定に戻ります。switchport コンフィギュレーション モードではコマンド プロンプトは（config-if）# と表示されます。
ステップ 4	**ipv6 snooping** [**attach-policy** *policy_name* [**vlan** {*vlan_id* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids*}] | **vlan** {*vlan_id* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] 例： (config-if)# **ipv6 snooping** or (config-if)# **ipv6 snooping attach-policy example_policy** or (config-if)# **ipv6 snooping vlan 111,112** or (config-if)# **ipv6 snooping attach-policy example_policy vlan 111,112**	インターフェイスまたはそのインターフェイス上の特定の VLAN にカスタム IPv6 スヌーピング ポリシーをアタッチします。デフォルト ポリシーをインターフェイスにアタッチするには、**attach-policy** キーワードを指定せずに **ipv6 snooping** コマンドを使用します。デフォルト ポリシーをインターフェイス上の VLAN にアタッチするには、**ipv6 snooping vlan** コマンドを使用します。デフォルト ポリシーは、セキュリティ レベル **guard**、デバイス ロール **node**、プロトコル **ndp** および **dhcp** です。
ステップ 5	**do show running-config** 例： #(config-if)# **do show running-config**	インターフェイス コンフィギュレーション モードを終了しないで、ポリシーが特定のインターフェイスにアタッチされていることを確認します。

IPv6 スヌーピング ポリシーをレイヤ 2 EtherChannel インターフェイスにアタッチする方法

EtherChannel インターフェイスまたは VLAN に IPv6 スヌーピング ポリシーをアタッチするには、特権 EXEC モードで次の手順を実行してください。

手順

	コマンドまたはアクション	目的
ステップ 1	**configure terminal** 例： `# 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 range** *Interface_name* 例： `(config)# interface range Po11`	EtherChannel の作成時に割り当てられたポート チャネル インターフェイスの名前を指定します。インターフェイス範囲コンフィギュレーション モードを開始します。 ヒント　インターフェイス名やタイプを簡単に参照するには **do show interfaces summary** コマンドを使用します。

ステップ 3	**ipv6 snooping** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
	例：
	`(config-if-range)# ipv6 snooping attach-policy example_policy`
	or
	`(config-if-range)# ipv6 snooping attach-policy example_policy vlan 222,223,224`
	or
	`(config-if-range)#ipv6 snooping vlan 222, 223,224`
		IPv6 スヌーピング ポリシーをインターフェイスまたはそのインターフェイス上の特定の VLAN にアタッチします。**attach-policy** オプションを使用しない場合、デフォルト ポリシーがアタッチされます。

ステップ 4	**do show running-config interface***portchannel_interface_name* 例： `#(config-if-range)# do show running-config int po11`	コンフィギュレーション モードを終了しないで、ポリシーが特定のインターフェイスにアタッチされていることを確認します。

IPv6スヌーピングポリシーをVLANにグローバルにアタッチする方法

複数のインターフェイス上の VLAN に IPv6 スヌーピング ポリシーをアタッチするには、特権 EXEC モードで次の手順を実行してください。

手順の概要

1. **configure terminal**
2. **vlan configuration** *vlan_list*
3. **ipv6 snooping** [**attach-policy** *policy_name*]
4. **do show running-config**

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 terminal** 例： `# 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	**vlan configuration** *vlan_list* 例： `(config)# vlan configuration 333`	VLAN インターフェイスのコンフィギュレーション モードを開始し、IPv6 スヌーピング ポリシーをアタッチする VLAN を指定します。
ステップ 3	**ipv6 snooping** [**attach-policy** *policy_name*] 例： `(config-vlan-config)#ipv6 snooping attach-policy example_policy`	すべてのスイッチおよびスタック インターフェイスで、IPv6 スヌーピング ポリシーを指定した VLAN にアタッチします。**attach-policy** オプションを使用しない場合、デフォルト ポリシーがアタッチされます。デフォルト ポリシーは、セキュリティ レベル **guard**、デバイス ロール **node**、プロトコル **ndp** および **dhcp** です。
ステップ 4	**do show running-config** 例： `#(config-if)# do show running-config`	インターフェイス コンフィギュレーション モードを終了しないで、ポリシーが特定の VLAN にアタッチされていることを確認します。

IPv6 バインディング テーブルの内容を設定する方法

IPv6 バインディング テーブル コンテンツを設定するには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **configure terminal**
2. [**no**] **ipv6 neighbor binding** [**vlan** *vlan-id* {*ipv6-address* **interface** interface_type *stack/module/port hw_address* [**reachable-lifetimevalue** [*seconds* | **default** | **infinite**] | [**tracking**{ [default | disable] [**reachable-lifetimevalue** [*seconds* | **default** | **infinite**] | [**enable** [**reachable-lifetimevalue** [*seconds* | **default** | **infinite**] | [**retry-interval** {*seconds*| **default** [**reachable-lifetimevalue** [*seconds* | **default** | **infinite**] }]
3. [**no**] **ipv6 neighbor binding max-entries** *number* [**mac-limit** *number* | **port-limit** *number* [**mac-limit** *number*] | **vlan-limit** *number* [[**mac-limit** *number*] | [**port-limit** *number* [**mac-limit***number*]]]]
4. **ipv6 neighbor binding logging**
5. **exit**
6. **show ipv6 neighbor binding**

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 terminal** 例： `# 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	[**no**] **ipv6 neighbor binding** [**vlan** *vlan-id* {*ipv6-address* **interface** interface_type *stack/module/port hw_address* [**reachable-lifetimevalue** [*seconds*	**default**	**infinite**]	[**tracking**{ [default	disable] [**reachable-lifetimevalue** [*seconds*	**default**	**infinite**]	[**enable** [**reachable-lifetimevalue** [*seconds*	**default**	**infinite**]	[**retry-interval** {*seconds*| **default** [**reachable-lifetimevalue** [*seconds*	**default**	**infinite**] }] 例： `(config)# ipv6 neighbor binding`	バインディング テーブル データベースにスタティック エントリを追加します。
ステップ 3	[**no**] **ipv6 neighbor binding max-entries** *number* [**mac-limit** *number*	**port-limit** *number* [**mac-limit** *number*]	**vlan-limit** *number* [[**mac-limit** *number*]	[**port-limit** *number* [**mac-limit***number*]]]] 例： `(config)# ipv6 neighbor binding max-entries 30000`	バインディング テーブル キャッシュに挿入できるエントリの最大数を指定します。
ステップ 4	**ipv6 neighbor binding logging** 例： `(config)# ipv6 neighbor binding logging`	バインディング テーブル メイン イベントのロギングを有効にします。
ステップ 5	**exit** 例：	グローバル コンフィギュレーション モードを終了して、ルータを特権 EXEC モードにします。

	コマンドまたはアクション	目的
	(config)# **exit**	
ステップ 6	**show ipv6 neighbor binding** 例： # **show ipv6 neighbor binding**	バインディング テーブルの内容を表示します。

IPv6 ネイバー探索検査ポリシーの設定方法

17.1.1 以降、IPv6 ND 検査機能は廃止され、SISF ベースのデバイス追跡に置き換えられます。対応する置き換えタスクについては、このドキュメントの「*SISF* ベースのデバイス追跡の設定」の章の「カスタム設定を使用したカスタムデバイス追跡ポリシーの作成」を参照してください。

特権 EXEC モードから、IPv6 ND 検査ポリシーを設定するには、次の手順に従ってください。

手順の概要

1. **configure terminal**
2. [**no**]**ipv6 nd inspection policy** *policy-name*
3. **device-role** {**host** | **switch**}
4. **limit address-count** *value*
5. **tracking** {**enable** [**reachable-lifetime** {*value* | **infinite**}] | **disable** [**stale-lifetime** {*value* | **infinite**}]}
6. **trusted-port**
7. **validate source-mac**
8. **no** {**device-role** | **limit address-count** | **tracking** | **trusted-port** | **validate source-mac**}
9. **default** {**device-role** | **limit address-count** | **tracking** | **trusted-port** | **validate source-mac**}
10. **do show ipv6 nd inspection policy** *policy_name*

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 terminal** 例： # **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	[**no**]**ipv6 nd inspection policy** *policy-name* 例： (config)# **ipv6 nd inspection policy example_policy**	ND 検査ポリシー名を指定し、ND 検査ポリシー 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	**device-role** {**host**	**switch**} 例： (config-nd-inspection)# **device-role switch**	ポートに接続されているデバイスの役割を指定します。デフォルトは **host** です。

	コマンドまたはアクション	目的
ステップ 4	**limit address-count** *value* 例： `(config-nd-inspection)# limit address-count 1000`	1 ～ 10,000 を入力します。
ステップ 5	**tracking** {**enable** [**reachable-lifetime** {*value* | **infinite**}] | **disable** [**stale-lifetime** {*value* | **infinite**}]} 例： `(config-nd-inspection)# tracking disable stale-lifetime infinite`	ポートのデフォルトのデバイス追跡ポリシーを上書きします。
ステップ 6	**trusted-port** 例： `(config-nd-inspection)# trusted-port`	信頼できるポートにするポートを設定します。
ステップ 7	**validate source-mac** 例： `(config-nd-inspection)# validate source-mac`	送信元 Media Access Control（MAC）アドレスをリンク層アドレスと照合します。
ステップ 8	**no** {**device-role** | **limit address-count** | **tracking** | **trusted-port** | **validate source-mac**} 例： `(config-nd-inspection)# no validate source-mac`	このコマンドの **no** 形式を使用してパラメータの現在の設定を削除します。
ステップ 9	**default** {**device-role** | **limit address-count** | **tracking** | **trusted-port** | **validate source-mac**} 例： `(config-nd-inspection)# default limit address-count`	設定をデフォルト値に戻します。
ステップ 10	**do show ipv6 nd inspection policy** *policy_name* 例： `(config-nd-inspection)# do show ipv6 nd inspection policy example_policy`	ND検査コンフィギュレーション モードを終了しないで ND 検査の設定を確認します。

IPv6 ネイバー探索検査ポリシーをインターフェイスにアタッチする方法

17.1.1 以降、IPv6 ND 検査機能は廃止され、SISF ベースのデバイス追跡に置き換えられます。対応する置き換えタスクについては、このドキュメントの「*SISF* ベースのデバイス追跡の設定」の章の「デバイス追跡ポリシーのインターフェイスへの適用」を参照してください。

インターフェイスまたはそのインターフェイス上の VLAN に IPv6 ND 検査ポリシーをアタッチするには、特権 EXEC モードで次の手順を実行してください。

手順の概要

1. **configure terminal**
2. **interface** Interface_type *stack/module/port*
3. **ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
4. **do show running-config**

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 terminal** 例： `# 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** Interface_type *stack/module/port* 例： `(config)# interface gigabitethernet 1/1/4`	インターフェイスのタイプおよび ID 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 3	**ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
	例：
	`(config-if)# ipv6 nd inspection attach-policy example_policy`
	or
	`(config-if)# ipv6 nd inspection attach-policy example_policy vlan 222,223,224`
	or
	`(config-if)# ipv6 nd inspection vlan 222, 223,224`
		ネイバー探索検査ポリシーをインターフェイスまたはそのインターフェイス上の特定の VLAN にアタッチします。**attach-policy** オプションを使用しない場合、デフォルト ポリシーがアタッチされます。

ステップ 4	**do show running-config** 例： `#(config-if)# do show running-config`	インターフェイス コンフィギュレーション モードを終了しないで、ポリシーが特定のインターフェイスにアタッチされていることを確認します。

IPv6 ネイバー探索検査ポリシーをレイヤ 2 EtherChannel インターフェイスにアタッチする方法

17.1.1 以降、IPv6 ND 検査機能は廃止され、SISF ベースのデバイス追跡機能に置き換えられます。対応する置き換えタスクについては、このドキュメントの「*SISF* ベースのデバイス追跡の設定」の章の「デバイス追跡ポリシーのインターフェイスへの適用」を参照してください。

EtherChannel インターフェイスまたは VLAN に IPv6 ネイバー探索検査ポリシーをアタッチするには、特権 EXEC モードで次の手順を実行してください。

手順の概要

1. **configure terminal**
2. **interface range** *Interface_name*
3. **ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
4. **do show running-config interface***portchannel_interface_name*

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 terminal** 例： `# 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 range** *Interface_name* 例： `(config)# interface Po11`	EtherChannel の作成時に割り当てられたポート チャネル インターフェイスの名前を指定します。インターフェイス範囲コンフィギュレーション モードを開始します。 ヒント　インターフェイス名やタイプを簡単に参照するには **do show interfaces summary** コマンドを使用します。

ステップ 3	**ipv6 nd inspection** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
	例：
	`(config-if-range)# ipv6 nd inspection attach-policy example_policy`
	or
	`(config-if-range)# ipv6 nd inspection attach-policy example_policy vlan 222,223,224`
		ND 検査ポリシーをインターフェイスまたはそのインターフェイス上の特定の VLAN にアタッチします。**attach-policy** オプションを使用しない場合、デフォルト ポリシーがアタッチされます。

	コマンドまたはアクション	目的
	or (config-if-range)#**ipv6 nd inspection vlan 222, 223,224**	
ステップ 4	**do show running-config interface***portchannel_interface_name* 例： #(config-if-range)# **do show running-config int po11**	コンフィギュレーション モードを終了しないで、ポリシーが特定のインターフェイスにアタッチされていることを確認します。

IPv6 ネイバー探索検査ポリシーを全体的に VLAN にアタッチする方法

17.1.1 以降、IPv6 ND 検査機能は廃止され、SISF ベースのデバイス追跡に置き換えられます。対応する置き換えタスクについては、このドキュメントの「*SISF* ベースのデバイス追跡の設定」の「デバイス追跡ポリシーの *VLAN* への適用」を参照してください。

複数のインターフェイス上の VLAN に IPv6 ND 探索ポリシーをアタッチするには、特権 EXEC モードで次の手順を実行してください。

手順の概要

1. **configure terminal**
2. **vlan configuration** *vlan_list*
3. **ipv6 nd inspection** [**attach-policy** *policy_name*]
4. **do show running-config**

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 terminal** 例： # **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	**vlan configuration** *vlan_list* 例： (config)# **vlan configuration 334**	VLAN インターフェイスのコンフィギュレーション モードを開始し、IPv6 スヌーピング ポリシーをアタッチする VLAN を指定します。
ステップ 3	**ipv6 nd inspection** [**attach-policy** *policy_name*] 例： (config-vlan-config)#**ipv6 nd inspection attach-policy example_policy**	すべてのスイッチおよびスタック インターフェイスで、IPv6 ネイバー探索ポリシーを指定した VLAN にアタッチします。**attach-policy** オプションを使用しない場合、デフォルト ポリシーがアタッチされます。

	コマンドまたはアクション	目的
		デフォルトのポリシーは、device-role **host**、no drop-unsecure、limit address-count disabled、sec-level minimum is disabled、tracking is disabled、no trusted-port、no validate source-mac です。
ステップ 4	**do show running-config** 例： `#(config-if)# do show running-config`	コンフィギュレーション モードを終了しないで、ポリシーが特定の VLAN にアタッチされていることを確認します。

IPv6 ルータ アドバタイズメント ガード ポリシーの設定方法

IPv6 ルータ アドバタイズメント ポリシーを設定するには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **configure terminal**
2. [**no**]**ipv6 nd raguard policy** *policy-name*
3. [**no**]**device-role** {**host** | **monitor** | **router** | **switch**}
4. [**no**]**hop-limit** {**maximum** | **minimum**} *value*
5. [**no**]**managed-config-flag** {**off** | **on**}
6. [**no**]**match** {**ipv6 access-list** *list* | **ra prefix-list** *list*}
7. [**no**]**other-config-flag** {**on** | **off**}
8. [**no**]**router-preference maximum** {**high** | **medium** | **low**}
9. [**no**]**trusted-port**
10. **default** {**device-role** | **hop-limit** {**maximum** | **minimum**} | **managed-config-flag** | **match** {**ipv6 access-list** | **ra prefix-list** } | **other-config-flag** | **router-preference maximum**| **trusted-port**}
11. **do show ipv6 nd raguard policy** *policy_name*

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 terminal** 例： `# 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	[**no**]**ipv6 nd raguard policy** *policy-name* 例： `(config)# ipv6 nd raguard policy example_policy`	RA ガード ポリシー名を指定し、RA ガード ポリシー 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 3	[**no**]**device-role** {**host** | **monitor** | **router** | **switch**} 例： `(config-nd-raguard)# device-role switch`	ポートに接続されているデバイスの役割を指定します。デフォルトは **host** です。 （注） ホスト側ポートとルータ側ポートの両方を備えたネットワークでは、ホスト側ポートまたは VLAN で **device-role host** を設定した RA ガードポリシーとともに、RA ガード機能が適切に動作できるように、ルータ側のポートで **device-role router** を設定した RA ガードポリシーを設定することが必須です。
ステップ 4	[**no**]**hop-limit** {**maximum** | **minimum**} *value* 例： `(config-nd-raguard)# hop-limit maximum 33`	（1～255）最大および最小のホップ制限値の範囲。 ホップ制限値によるルータアドバタイズメントメッセージのフィルタリングを有効にします。不正RA メッセージは低いホップ制限値（IPv4 の Time to Live と同じ）を持つ可能性があるため、ホストによって受け入れられると、ホストが不正 RA メッセージ ジェネレータを超えて宛先にトラフィックを生成することができなくなります。指定されていないホップ制限値を持つRAメッセージはブロックされます。 設定されていない場合、このフィルタは無効になります。「**minimum**」を設定して、指定する値より低いホップ制限値を持つRAメッセージをブロックします。「**maximum**」を設定して、指定する値より高いホップ制限値を持つRAメッセージをブロックします。
ステップ 5	[**no**]**managed-config-flag** {**off** | **on**} 例： `(config-nd-raguard)# managed-config-flag on`	管理アドレス設定（「M」フラグ）フィールドに基づいてルータアドバタイズメントメッセージのフィルタリングを有効にします。「M」フィールドが 1 の不正 RA メッセージの結果としてホストが不正 DHCPv6 サーバを使用する場合があります。設定されていない場合、このフィルタは無効になります。 **On**：「M」値が 1 の RA メッセージを受け入れて転送し、0 のものをブロックします。 **Off**：「M」値が 0 の RA メッセージを受け入れて転送し、1 のものをブロックします。
ステップ 6	[**no**]**match** {**ipv6 access-list** *list* | **ra prefix-list** *list*} 例：	指定したプレフィックス リストまたはアクセス リストと照合します。

	コマンドまたはアクション	目的
	`(config-nd-raguard)# match ipv6 access-list example_list`	
ステップ 7	**[no]other-config-flag** {**on** | **off**} 例： `(config-nd-raguard)# other-config-flag on`	その他の設定（「O」フラグ）フィールドに基づくルータ アドバタイズメント メッセージのフィルタリングを有効にします。「O」フィールドが 1 の不正 RA メッセージの結果としてホストが不正 DHCPv6 サーバを使用する場合があります。設定されていない場合、このフィルタは無効になります。 **On**：「O」値が 1 の RA メッセージを受け入れて転送し、0 のものをブロックします。 **Off**：「O」値が 0 の RA メッセージを受け入れて転送し、1 のものをブロックします。
ステップ 8	**[no]router-preference maximum** {**high** | **medium** | **low**} 例： `(config-nd-raguard)# router-preference maximum high`	「Router Preference」フラグを使用したルータ アドバタイズメント メッセージのフィルタリングを有効にします。設定されていない場合、このフィルタは無効になります。 • **high**：「Router Preference」が「high」、「medium」、または「low」に設定された RA メッセージを受け入れます。 • **medium**：「Router Preference」が「high」に設定された RA メッセージをブロックします。 • **low**：「Router Preference」が「medium」または「high」に設定された RA メッセージをブロックします。
ステップ 9	**[no]trusted-port** 例： `(config-nd-raguard)# trusted-port`	信頼できるポートとして設定すると、すべての接続デバイスが信頼され、より詳細なメッセージ検証は実行されません。
ステップ 10	**default** {**device-role** | **hop-limit** {**maximum** | **minimum**} | **managed-config-flag** | **match** {**ipv6 access-list** | **ra prefix-list** } | **other-config-flag** | **router-preference maximum**| **trusted-port**} 例： `(config-nd-raguard)# default hop-limit`	コマンドをデフォルト値に戻します。
ステップ 11	**do show ipv6 nd raguard policy** *policy_name* 例： `(config-nd-raguard)# do show ipv6 nd raguard policy example_policy`	（任意）：RA ガード ポリシー コンフィギュレーション モードを終了しないで ND ガード ポリシー設定を表示します。

IPv6 ルータ アドバタイズメント ガード ポリシーをインターフェイスにアタッチする方法

インターフェイスまたはそのインターフェース上の VLAN に IPv6 ルータ アドバタイズメント ポリシーをアタッチするには、特権 EXEC モードで次の手順を実行してください。

手順の概要

1. **configure terminal**
2. **interface** Interface_type *stack/module/port*
3. **ipv6 nd raguard** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
4. **do show running-config**

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 terminal** 例： `# 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** Interface_type *stack/module/port* 例： `(config)# interface gigabitethernet 1/1/4`	インターフェイスのタイプおよび ID 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 3	**ipv6 nd raguard** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
	例：
	`(config-if)# ipv6 nd raguard attach-policy example_policy`
	or
	`(config-if)# ipv6 nd raguard attach-policy example_policy vlan 222,223,224`
	or
	`(config-if)# ipv6 nd raguard vlan 222, 223,224`
		ネイバー探索検査ポリシーをインターフェイスまたはそのインターフェイス上の特定の VLAN にアタッチします。**attach-policy** オプションを使用しない場合、デフォルト ポリシーがアタッチされます。

ステップ 4	**do show running-config** 例： `#(config-if)# do show running-config`	コンフィギュレーション モードを終了しないで、ポリシーが特定のインターフェイスにアタッチされていることを確認します。

IPv6ルータアドバタイズメントガードポリシーをレイヤ2EtherChannelインターフェイスにアタッチする方法

EtherChannel インターフェイスまたは VLAN に IPv6 ルータ アドバタイズメント ガード ポリシーをアタッチするには、特権 EXEC モードで次の手順を実行してください。

手順の概要

1. **configure terminal**
2. **interface range** *Interface_name*
3. **ipv6 nd raguard** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
4. **do show running-config interface***portchannel_interface_name*

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 terminal** 例： `# 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 range** *Interface_name* 例： `(config)# interface Po11`	EtherChannel の作成時に割り当てられたポート チャネル インターフェイスの名前を指定します。インターフェイス範囲コンフィギュレーション モードを開始します。 ヒント　インターフェイス名やタイプを簡単に参照するには do show interfaces summary コマンドを使用します。

ステップ 3	**ipv6 nd raguard** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
	例：
	`(config-if-range)# ipv6 nd raguard attach-policy example_policy`
	or
	`(config-if-range)# ipv6 nd raguard attach-policy example_policy vlan 222,223,224`
	or
	`(config-if-range)#ipv6 nd raguard vlan 222, 223,224`
		RA ガード ポリシーをインターフェイスまたはそのインターフェイス上の特定の VLAN にアタッチします。**attach-policy** オプションを使用しない場合、デフォルト ポリシーがアタッチされます。

	コマンドまたはアクション	目的
ステップ 4	**do show running-config interface***portchannel_interface_name* 例： `#(config-if-range)# do show running-config int po11`	コンフィギュレーション モードを終了しないで、ポリシーが特定のインターフェイスにアタッチされていることを確認します。

IPv6 ルータ アドバタイズメント ガード ポリシーを VLAN にグローバルにアタッチする方法

インターフェイスに関係なく VLAN に IPv6 ルータ アドバタイズメント ポリシーをアタッチするには、特権 EXEC モードで次の手順を実行してください。

手順の概要

1. **configure terminal**
2. **vlan configuration** *vlan_list*
3. **ipv6 dhcp guard** [**attach-policy** *policy_name*]
4. **do show running-config**

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 terminal** 例： `# 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	**vlan configuration** *vlan_list* 例： `(config)# vlan configuration 335`	VLAN インターフェイスのコンフィギュレーション モードを開始し、IPv6 RA ガード ポリシーをアタッチする VLAN を指定します。
ステップ 3	**ipv6 dhcp guard** [**attach-policy** *policy_name*] 例： `(config-vlan-config)#ipv6 nd raguard attach-policy example_policy`	すべてのスイッチおよびスタック インターフェイスで、IPv6 RA ガード ポリシーを指定した VLAN にアタッチします。**attach-policy** オプションを使用しない場合、デフォルト ポリシーがアタッチされます。
ステップ 4	**do show running-config** 例： `#(config-if)# do show running-config`	コンフィギュレーション モードを終了しないで、ポリシーが特定の VLAN にアタッチされていることを確認します。

IPv6 DHCP ガード ポリシーの設定方法

IPv6 DHCP（DHCPv6）ガード ポリシーを設定するには、特権 EXEC モードで次の手順を実行します。

手順の概要

1. **configure terminal**
2. [**no**]**ipv6 dhcp guard policy** *policy-name*
3. [**no**]**device-role** {**client** | **server**}
4. [**no**] **match server access-list** *ipv6-access-list-name*
5. [**no**] **match reply prefix-list** *ipv6-prefix-list-name*
6. [**no**]**preference**{ **max** *limit* | **min** *limit* }
7. [**no**] **trusted-port**
8. **default** {**device-role** | **trusted-port**}
9. **do show ipv6 dhcp guard policy** *policy_name*

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 terminal** 例： `# 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	[**no**]**ipv6 dhcp guard policy** *policy-name* 例： `(config)# ipv6 dhcp guard policy example_policy`	DHCPv6 ガード ポリシー名を指定し、DHCPv6 ガード ポリシー 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	[**no**]**device-role** {**client** | **server**} 例： `(config-dhcp-guard)# device-role server`	（任意）特定の役割のデバイスからのものではないポート上の DHCPv6 応答および DHCPv6 アドバタイズメントをフィルタします。デフォルトは **client** です。 • **client**：デフォルト値。アタッチされたデバイスがクライアントであることを指定します。サーバ メッセージにはこのポートで破棄されます。 • **server**：適用されたデバイスが DHCPv6 サーバであることを指定します。このポートでは、サーバ メッセージが許可されます。
ステップ 4	[**no**] **match server access-list** *ipv6-access-list-name* 例： `;;Assume a preconfigured IPv6 Access List as`	（任意）。アドバタイズされた DHCPv6 サーバまたはリレー アドレスが認証されたサーバのアクセス リストからのものであることの確認を有効にします（アクセス リストの宛先アドレスは「any」です）。

	コマンドまたはアクション	目的
	follows: (config)# **ipv6 access-list my_acls** (config-ipv6-acl)# **permit host FE80::A8BB:CCFF:FE01:F700 any** ;;configure DCHPv6 Guard to match approved access list. (config-dhcp-guard)# **match server access-list my_acls**	設定されていない場合、このチェックは回避されます。空のアクセス リストは、permit all として処理されます。
ステップ 5	[**no**] **match reply prefix-list** *ipv6-prefix-list-name* 例： ;;Assume a preconfigured IPv6 prefix list as follows: (config)# **ipv6 prefix-list my_prefix permit 2001:0DB8::/64 le 128** ;; Configure DCHPv6 Guard to match prefix (config-dhcp-guard)# **match reply prefix-list my_prefix**	（任意）DHCPv6 応答メッセージ内のアドバタイズされたプレフィクスが設定された承認プレフィクス リストからのものであることの確認を有効にします。設定されていない場合、このチェックは回避されます。空のプレフィクス リストは、permit として処理されます。
ステップ 6	[**no**]**preference**{ **max** *limit* | **min** *limit* } 例： (config-dhcp-guard)# **preference max 250** (config-dhcp-guard)#**preference min 150**	**device-role** が **server** である場合に **max** および **min** を設定して、DHCPv6 サーバ アドバタイズメント値をサーバ優先度値に基づいてフィルタします。デフォルトではすべてのアドバタイズメントが許可されます。 **max** *limit*：（0 ～ 255）（任意）アドバタイズされたプリファレンス（[preference] オプション内）が指定された制限未満であるかどうかの検証を有効にします。デフォルトは 255 です。設定されていない場合、このチェックは回避されます。 **min** *limit*：（0 ～ 255）（任意）アドバタイズされたプリファレンス（[preference] オプション内）が指定された制限を超過しているかどうかの検証を有効にします。デフォルトは 0 です。設定されていない場合、このチェックは回避されます。
ステップ 7	[**no**] **trusted-port** 例： (config-dhcp-guard)# **trusted-port**	（任意）**trusted-port**：ポートを信頼モードに設定します。このポートでは、これ以上のポリシングは実行されません。 （注） 信頼できるポートを設定した場合、device-role オプションは使用できません。
ステップ 8	**default** {**device-role** | **trusted-port**} 例： (config-dhcp-guard)# **default device-role**	（任意）**default**：コマンドをデフォルトに設定します。

	コマンドまたはアクション	目的
ステップ 9	**do show ipv6 dhcp guard policy** *policy_name* 例： `(config-dhcp-guard)# do show ipv6 dhcp guard policy example_policy`	（任意）コンフィギュレーション サブモードを終了せずに IPv6 DHCP のガード ポリシーの設定を表示します。*policy_name* 変数を省略すると、すべての DHCPv6 ポリシーが表示されます。

DHCPv6 ガード設定の例

```
enable
configure terminal
ipv6 access-list acl1
 permit host FE80::A8BB:CCFF:FE01:F700 any
ipv6 prefix-list abc permit 2001:0DB8::/64 le 128
ipv6 dhcp guard policy pol1
 device-role server
 match server access-list acl1
 match reply prefix-list abc
 preference min 0
 preference max 255
 trusted-port
interface GigabitEthernet 0/2/0
 switchport
 ipv6 dhcp guard attach-policy pol1 vlan add 1
 vlan 1
  ipv6 dhcp guard attach-policy pol1
show ipv6 dhcp guard policy pol1
```

IPv6 DHCP ガード ポリシーを全体的に VLAN にアタッチする方法

複数のインターフェイス上の VLAN に IPv6 DHCP のガード ポリシーをアタッチするには、特権 EXEC モードで次の手順を実行してください。

手順の概要

1. **configure terminal**
2. **vlan configuration** *vlan_list*
3. **ipv6 dhcp guard** [**attach-policy** *policy_name*]
4. **do show running-config**

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 terminal** 例： `# 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。

	コマンドまたはアクション	目的
ステップ 2	**vlan configuration** *vlan_list* 例： `(config)# vlan configuration 334`	VLAN インターフェイスのコンフィギュレーションモードを開始し、IPv6 スヌーピング ポリシーをアタッチする VLAN を指定します。
ステップ 3	**ipv6 dhcp guard** [**attach-policy** *policy_name*] 例： `(config-vlan-config)#ipv6 dhcp guard attach-policy example_policy`	すべてのスイッチおよびスタック インターフェイスで、IPv6 ネイバー探索ポリシーを指定した VLAN にアタッチします。**attach-policy** オプションを使用しない場合、デフォルト ポリシーがアタッチされます。デフォルト ポリシーは、device-role **client**、**no** trusted-port です。
ステップ 4	**do show running-config** 例： `#(config-if)# do show running-config`	コンフィギュレーションモードを終了しないで、ポリシーが特定の VLAN にアタッチされていることを確認します。

IPv6 DHCP ガードポリシーをレイヤ 2 EtherChannel インターフェイスにアタッチする方法

EtherChannel インターフェイスまたは VLAN に IPv6 DHCP ガード ポリシーをアタッチするには、特権 EXEC モードで次の手順を実行してください。

手順の概要

1. **configure terminal**
2. **interface range** *Interface_name*
3. **ipv6 dhcp guard** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
4. **do show running-config interface***portchannel_interface_name*

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 terminal** 例： `# 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 range** *Interface_name* 例： `(config)# interface Po11`	EtherChannel の作成時に割り当てられたポート チャネル インターフェイスの名前を指定します。インターフェイス範囲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
		ヒント　インターフェイス名やタイプを簡単に参照するには **do show interfaces summary** コマンドを使用します。

ステップ 3	**ipv6 dhcp guard** [**attach-policy** *policy_name* [**vlan** {*vlan_ids* | **add** *vlan_ids* | **except** *vlan_ids* | **none** | **remove** *vlan_ids* | **all**}] | **vlan** [{*vlan_ids* | **add** *vlan_ids* | **except***vlan_ids* | **none** | **remove** *vlan_ids* | **all**}]
	例：
	`(config-if-range)# ipv6 dhcp guard attach-policy example_policy`
	or
	`(config-if-range)# ipv6 dhcp guard attach-policy example_policy vlan 222,223,224`
	or
	`(config-if-range)#ipv6 dhcp guard vlan 222, 223,224`
		DHCP ガードポリシーをインターフェイスまたはそのインターフェイス上の特定の VLAN にアタッチします。**attach-policy** オプションを使用しない場合、デフォルト ポリシーがアタッチされます。

ステップ 4	**do show running-config interface***portchannel_interface_name* 例： `#(config-if-range)# do show running-config int po11`	コンフィギュレーションモードを終了しないで、ポリシーが特定のインターフェイスにアタッチされていることを確認します。

IPv6 DHCP ガード ポリシーを全体的に VLAN にアタッチする方法

複数のインターフェイス上の VLAN に IPv6 DHCP のガード ポリシーをアタッチするには、特権 EXEC モードで次の手順を実行してください。

手順の概要

1. **configure terminal**
2. **vlan configuration** *vlan_list*
3. **ipv6 dhcp guard** [**attach-policy** *policy_name*]
4. **do show running-config**

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 terminal** 例：	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。

	コマンドまたはアクション	目的
	`# configure terminal`	
ステップ 2	**vlan configuration** *vlan_list* 例： `(config)# vlan configuration 334`	VLAN インターフェイスのコンフィギュレーション モードを開始し、IPv6 スヌーピング ポリシーをアタッチする VLAN を指定します。
ステップ 3	**ipv6 dhcp guard** [**attach-policy** *policy_name*] 例： `(config-vlan-config)#ipv6 dhcp guard attach-policy example_policy`	すべてのスイッチおよびスタック インターフェイスで、IPv6 ネイバー探索ポリシーを指定した VLAN にアタッチします。**attach-policy** オプションを使用しない場合、デフォルト ポリシーがアタッチされます。デフォルト ポリシーは、device-role **client**、**no** trusted-port です。
ステップ 4	**do show running-config** 例： `#(config-if)# do show running-config`	コンフィギュレーション モードを終了しないで、ポリシーが特定の VLAN にアタッチされていることを確認します。

IPv6 ソース ガードの設定方法

手順の概要

1. **configure terminal**
2. [**no**] **ipv6 source-guard policy** *policy_name*
3. [**deny global-autoconf**] [**permit link-local**] [**default**{. . . }] [**exit**] [**no**{. . . }]
4. **end**
5. **show ipv6 source-guard policy** *policy_name*

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 terminal** 例： `# 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	[**no**] **ipv6 source-guard policy** *policy_name* 例： `(config)# ipv6 source-guard policy example_policy`	IPv6 ソース ガード ポリシー名を指定し、IPv6 ソース ガード ポリシー 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	[**deny global-autoconf**] [**permit link-local**] [**default**{. . . }] [**exit**] [**no**{. . . }] 例：	（任意）IPv6 ソース ガード ポリシーを定義します。 • **deny global-autoconf**：自動設定されたグローバル アドレスからのデータ トラフィックを拒否

	コマンドまたはアクション	目的
	(config-sisf-sourceguard)# **deny global-autoconf**	します。これは、リンク上のすべてのグローバルアドレスがDHCPによって割り当てられている際に、管理者が、自己設定されたアドレスを持つホストによるトラフィックの送信をブロックしたい場合に役立ちます。 • **permit link-local**：リンクローカルアドレスから送信されたすべてのデータ トラフィックを許可します。 （注） ソース ガード ポリシーでは trusted オプションはサポートされません。
ステップ 4	**end** 例： (config-sisf-sourceguard)# **end**	IPv6 ソース ガード ポリシー 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	**show ipv6 source-guard policy** *policy_name* 例： # **show ipv6 source-guard policy example_policy**	ポリシー設定と、そのポリシーが適用されるすべてのインターフェイスを表示します。

次のタスク

インターフェイスに IPv6 ソース ガード ポリシーを適用します。

IPv6 ソース ガード ポリシーをインターフェイスにアタッチする方法

手順の概要

1. **configure terminal**
2. **interface** Interface_type *stack/module/port*
3. **ipv6 source-guard**［**attach-policy** <*policy_name*> ］
4. **show ipv6 source-guard policy** *policy_name*

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 terminal** 例： # **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。

	コマンドまたはアクション	目的
ステップ 2	**interface** Interface_type *stack/module/port* 例： `(config)# interface gigabitethernet 1/1/4`	インターフェイスのタイプおよび ID 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 3	**ipv6 source-guard** [**attach-policy** <*policy_name*>] 例： `(config-if)# ipv6 source-guard attach-policy example_policy`	インターフェイスに IPv6 ソース ガード ポリシーをアタッチします。**attach-policy** オプションを使用しない場合、デフォルト ポリシーがアタッチされます。
ステップ 4	**show ipv6 source-guard policy** *policy_name* 例： `#(config-if)# show ipv6 source-guard policy example_policy`	ポリシー設定と、そのポリシーが適用されるすべてのインターフェイスを表示します。

IPv6 ソース ガード ポリシーをレイヤ 2 EtherChannel インターフェイスにアタッチする方法

手順の概要

1. **configure terminal**
2. **interface port-channel** *port-channel-number*
3. **ipv6 source-guard** [**attach-policy** <*policy_name*>]
4. **show ipv6 source-guard policy** *policy_name*

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 terminal** 例： `# 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 port-channel** *port-channel-number* 例： `(config)# interface Po4`	インターフェイスのタイプとポート番号を指定し、スイッチをポート チャネル コンフィギュレーション モードにします。
ステップ 3	**ipv6 source-guard** [**attach-policy** <*policy_name*>] 例： `(config-if) # ipv6 source-guard attach-policy example_policy`	インターフェイスに IPv6 ソース ガード ポリシーをアタッチします。**attach-policy** オプションを使用しない場合、デフォルト ポリシーがアタッチされます。

	コマンドまたはアクション	目的
ステップ 4	**show ipv6 source-guard policy** *policy_name* 例： `(config-if) #show ipv6 source-guard policy example_policy`	ポリシー設定と、そのポリシーが適用されるすべてのインターフェイスを表示します。

IPv6 プレフィックス ガードの設定方法

（注） プレフィックス ガードが適用されている場合にリンクローカル アドレスから送信されたルーティング プロトコル制御パケットを許可するには、ソースガード ポリシー コンフィギュレーション モードで permit link-local コマンドを有効にします。

手順の概要

1. ［**no**］**ipv6 source-guard policy** *source-guard-policy*
2. ［ **no** ］**validate address**
3. **validate prefix**
4. **exit**
5. **show ipv6 source-guard policy**［*source-guard-policy*］

手順の詳細

	コマンドまたはアクション	目的
ステップ 1	［**no**］**ipv6 source-guard policy** *source-guard-policy* 例： `(config)# ipv6 source-guard policy my_snooping_policy`	IPv6 ソースガードポリシー名を定義して、スイッチ統合セキュリティ機能のソースガード ポリシー 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** ］**validate address** 例： `(config-sisf-sourceguard)# no validate address`	アドレス検証機能を無効にし、IPv6プレフィックスガード機能を設定できるようにします。
ステップ 3	**validate prefix** 例： `(config-sisf-sourceguard)# validate prefix`	IPv6 プレフィックスガード動作を実行するよう、IPv6 ソースガードを有効にします。
ステップ 4	**exit** 例： `(config-sisf-sourceguard)# exit`	スイッチ統合セキュリティ機能のソースガードポリシー 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。

	コマンドまたはアクション	目的
ステップ 5	**show ipv6 source-guard policy**［*source-guard-policy*］ 例： `# show ipv6 source-guard policy policy1`	IPv6 ソースガード ポリシー設定を表示します。

IPv6プレフィックスガードポリシーをインターフェイスにアタッチする方法

手順の概要

1. **configure terminal**
2. **interface** Interface_type *stack/module/port*
3. **ipv6 source-guard attach-policy** *policy_name*
4. **show ipv6 source-guard policy** *policy_name*

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 terminal** 例： `# 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** Interface_type *stack/module/port* 例： `(config)# interface gigabitethernet 1/1/4`	インターフェイスのタイプおよび ID 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 3	**ipv6 source-guard attach-policy** *policy_name* 例： `(config-if)# ipv6 source-guard attach-policy example_policy`	インターフェイスに IPv6 ソース ガード ポリシーをアタッチします。**attach-policy** オプションを使用しない場合、デフォルト ポリシーがアタッチされます。
ステップ 4	**show ipv6 source-guard policy** *policy_name* 例： `(config-if)# show ipv6 source-guard policy example_policy`	ポリシー設定と、そのポリシーが適用されるすべてのインターフェイスを表示します。

IPv6 プレフィックス ガード ポリシーをレイヤ 2 EtherChannel インターフェイスにアタッチする方法

手順の概要

1. **configure terminal**
2. **interface port-channel** *port-channel-number*
3. **ipv6 source-guard**［**attach-policy** <*policy_name*> ］
4. **show ipv6 source-guard policy** *policy_name*

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 terminal** 例： `# 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 port-channel** *port-channel-number* 例： `(config)# interface Po4`	インターフェイスのタイプとポート番号を指定し、スイッチをポート チャネル コンフィギュレーション モードにします。
ステップ 3	**ipv6 source-guard**［**attach-policy** <*policy_name*> ］ 例： `(config-if)# ipv6 source-guard attach-policy example_policy`	インターフェイスに IPv6 ソース ガード ポリシーをアタッチします。**attach-policy** オプションを使用しない場合、デフォルト ポリシーがアタッチされます。
ステップ 4	**show ipv6 source-guard policy** *policy_name* 例： `(config-if)# show ipv6 source-guard policy example_policy`	ポリシー設定と、そのポリシーが適用されるすべてのインターフェイスを表示します。

IPv6 ファースト ホップ セキュリティの設定例

例：IPv6 ソース ガード ポリシーをレイヤ 2 EtherChannel インターフェイスにアタッチする方法

次の例は、IPv6 ソース ガード ポリシーをレイヤ 2 EtherChannel インターフェイスにアタッチする方法を示しています。

```
Switch# configure terminal
Switch(config)# ipv6 source-guard policy POL
Switch(config-sisf-sourceguard) # validate address
switch(config-sisf-sourceguard)# exit
```

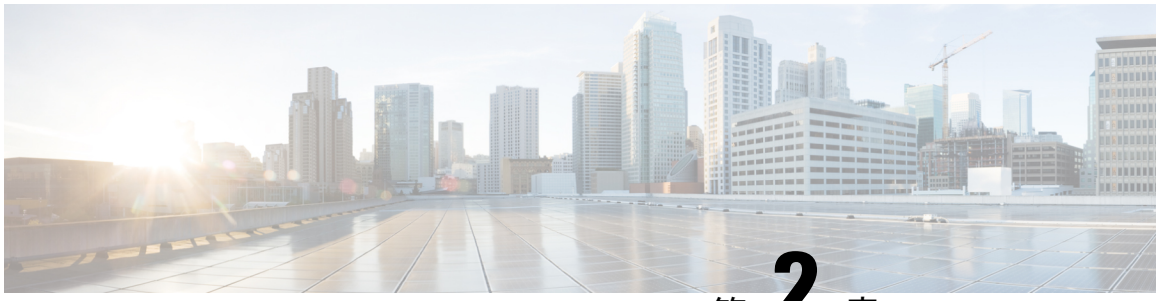
```
Switch(config)# interface Po4
Switch(config)# ipv6 snooping
Switch(config-if)# ipv6 source-guard attach-policy POL
Switch(config-if)# exit
switch(config)#
```

例：IPv6 プレフィックス ガード ポリシーをレイヤ 2 EtherChannel インターフェイスにアタッチする方法

次の例は、IPv6 プレフィックス ガード ポリシーをレイヤ 2 EtherChannel インターフェイスにアタッチする方法を示しています。

```
Switch# configure terminal
Switch(config)# ipv6 source-guard policy POL
Switch (config-sisf-sourceguard)# no validate address
Switch((config-sisf-sourceguard)# validate prefix
Switch(config)# interface Po4
Switch(config-if)# ipv6 snooping
Switch(config-if)# ipv6 source-guard attach-policy POL
```

例：IPv6 プレフィックス ガード ポリシーをレイヤ 2 EtherChannel インターフェイスにアタッチする方法



第 2 章

Cisco TrustSec VRF 対応 SGT

VRF-Aware SXP

仮想ルーティングおよびフォワーディング（VRF）のセキュリティグループタグ（SGT）Exchange Protocol（SXP）の実装は、特定の VRF と SXP 接続をバインドします。Cisco TrustSec を有効にする前に、ネットワーク トポロジがレイヤ 2 またはレイヤ 3 の VPN に対して正しく設定されており、すべての VRF が設定されていることを前提としています。

SXP VRF サポートは、次のようにまとめることができます。

- 1 つの VRF には 1 つの SXP 接続のみをバインドできます。
- 別の VRF が重複する SXP ピアまたは送信元 IP アドレス持つ可能性があります。
- 1 つの VRF で学習（追加または削除）された IP-SGT マッピングは、同じ VRF ドメインでのみ更新できます。SXP 接続は異なる VRF にバインドされたマッピングを更新できません。SXP 接続が VRF で終了しない場合は、その VRF の IP-SGT マッピングは SXP によって更新されません。
- VRF ごとに複数のアドレス ファミリがサポートされています。そのため、VRF ドメインの 1 つの SXP 接続が IPV4 および IPV6 両方の IP-SGT マッピングを転送できます。
- SXP には VRF あたりの接続数および IP-SGT マッピング数の制限はありません。

VRF 対応 SGT および SGACL の IPv6 サポート

Cisco IOS XE Bengaluru 17.6.x リリース以降では、VRF 対応セキュリティグループタグ（SGT）および SG アクセスコントロールリスト（SGACL）で IPv6 がサポートされています。この機能は、IPv4 の場合と同じ機能を IPv6 に対して拡張します。

SGT および SGACL 機能に対する IPv6 サポートにより、次の機能が有効になります。

- SGT バインディング
  - SGT への IPv6 アドレス間の静的バインディング
  - VLAN から SGT へのバインディング
  - IPv6 アドレスと SGT 間のマッピングの動的学習
- 施行
  - UDP または TCP ポートに基づく IPv6 トラフィックに対する SGACL 適用
  - 上位層プロトコルタイプに基づく IPv6 トラフィックに対する SGACL 適用

（注）
- SGT バインディングは、リンクローカルアドレスではサポートされていません。
- SGACL はマルチキャストトラフィックには適用されません。

IPv6 SGT と SGACL のスケール値は IPv4 と IPv6 の両方で同じであり、ほとんどの CLI コマンドは変更されていません。

IPv6 サポートの詳細については、次の項を参照してください。

- IPv4 と IPv6 が SGT および SGACL テーブルを共有する方法 （36 ページ）
- SGT および SGACL スケール値 （37 ページ）

Cisco.com の Cisco TrustSec コンフィギュレーション ガイド、Cisco IOS XE 17 [英語] も参照してください。

IPv4 と IPv6 が SGT および SGACL テーブルを共有する方法

IPv4 と IPv6 は、FPGA で SGT および SGACL テーブルを共有します。次のリストに、共有の管理方法を示します。

- IPv4 または IPv6 を有効にすると、設定に基づいてテーブル全体が使用されます。
- IPv4 と IPv6 を有効にすると、最初の要求を行う機能に基づいてテーブルが共有されます。

- SGT および SGACL テーブルの上限を超えると、適切な syslog が生成されます。
- サポートされていないポリシーを設定すると、適切な syslog が生成されます。

SGT および SGACL スケール値

次の表に、IPv4 と IPv6 のスケール値を示します。

エントリの種類	スケール値	説明
ホストから SGT	1024	ホストから SGT へのバインド
サブネットから SGT	64	ネットワークから SGT へのバインド
SGT X DGT マトリックス	21 X 21	SGT から DGT へのマッピング
SGACL ポリシーリストサイズ	15	各 SGACL の最大 ACE
ロギングカウンタ [31:0] SGT および DGT ペアの数	32	最大ペア数

（注） デフォルトでは、ロギングは 32 の SGT と DGT のペアに対してのみ有効になっています。ただし、ロギングを有効にするペアを指定できます。32 ペアのうち任意のペアに対するロギングを無効にし、異なるペアのロギングを有効にできます。

- ロギングが有効になっている SGT と DGT のペアを表示するには、show platform hardware cts cell-logging コマンドを使用します。
- 特定の SGT および DGT ペアのロギングを無効にするには、no platform cts logging コマンドを使用します。
- 特定の SGT と DGT のペアのロギングを有効にするには、platform cts logging コマンドを使用します。

次のテキストは、no platform cts logging コマンドのオプションを示しています。

```
Device> enable
Device#configure terminal
Device(config)#no platform cts logging ?
all Disable logging for all the cells
default default logging list
from Source Group Tag (SGT) for enabling logging
```

Cisco TrustSec VRF 対応 SGT の設定方法

このセクションでは、Cisco TrustSec VRF 対応 SGT の設定方法について説明します。

VRF と SGT のマッピングの設定

手順の概要

1. **enable**
2. **configure terminal**
3. **cts role-based sgt-map vrf** *vrf-name* {**ip4_netaddress** | *ipv6_netaddress* | **host** {*ip4_address* | *ip6_address*}}] **sgt** *sgt_number*
4. **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	**cts role-based sgt-map vrf** *vrf-name* {**ip4_netaddress** | *ipv6_netaddress* | **host** {*ip4_address* | *ip6_address*}}] **sgt** *sgt_number*
	例：
	`Device(config)# cts role-based sgt-map vrf red 10.0.0.3 sgt 23`
	例：
	`Device(config)# cts role-based sgt-map vrf VRF_1 2405:201:c::f115 sgt 1201`
		指定された VRF のパケットに SGT を適用します。
		IP-SGT バインドは、指定された VRF と、IP アドレスのタイプによって示される IP プロトコルのバージョンに関連付けられた IP-SGT のテーブルに入力されます。

ステップ 4	**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 モードに戻ります。

Cisco TrustSec VRF 対応 SGT の設定例

このセクションでは、Cisco TrustSec VRF 対応 SGT の設定例を示します。

例：VRF と SGT のマッピングの設定

IPv4 の例：

```
Device> enable
Device# configure terminal
Device(config)# cts role-based sgt-map vrf VRF_1 22.1.1.1 sgt 1204
Device(config)# end
```

IPv6 の例：

```
Device> enable
Device# configure terminal
Device(config)# cts role-based sgt-map vrf VRF_1 2405:201:c::f115 sgt 1201
Device(config)# end
```

例：ロールベース アクセス リスト コマンド

```
Switch(config)# ipv6 access-list role-based acl-name
Switch(config-rb-acl)#?
Role-based Access List configuration commands:
<1-2147483647> Sequence Number
default    Set a command to its defaults
deny       Specify packets to reject
exit       Exit from access-list configuration mode
no         Negate a command or set its defaults
permit     Specify packets to forward
remark     Access list entry comment
Switch(config-rb-acl)#
```

ACE ポート範囲

Cisco IOS XE Bengaluru 17.6.x リリース以降、TrustSec FPGA モジュールは、いくつかのスケーリングの問題に対処するためにポリシー要素のポート範囲オプションをサポートしています。

FPGA モジュールは、TrustSec の一部として IP-to-SGT バインディングと SGACL ポリシーを維持します。Cisco IE3400 スイッチは、各セルで 21 X 21 SGT または DGT ペアと 15 個のポリシーをサポートし、IP プロトコルフィールド、L4 送信元ポート、および L4 宛先ポートを照合します。

ただし、一致基準を指定すると、ユーザアクセス権限を拡張できない場合があります。そのため、TrustSec FPGA モジュールは、各セルでサポートされるポリシーを15 個に維持することで、各ポリシー要素のポート範囲オプションをサポートします。

この機能強化により、次のリストに示すように複数のルールを組み合わせることができます。

- IP プロトコルフィールドの照合
- L4 送信元開始ポートと終了ポートの照合
- L4 宛先開始ポートと終了ポートの照合

例：ACE ポート範囲のロールベース アクセス リスト コマンド

次のコマンドを使用して、送信元ポートと宛先ポートの ACE ポート範囲を設定できます。

```
Switch(config)# ip access-list role-based rbacl
Switch(config-rb-acl)#10 deny tcp dst range ftp-data telnet
Switch(config-rb-acl)#20 permit tcp dst lt 10
Switch(config-rb-acl)#30 deny tcp dst gt 50
```

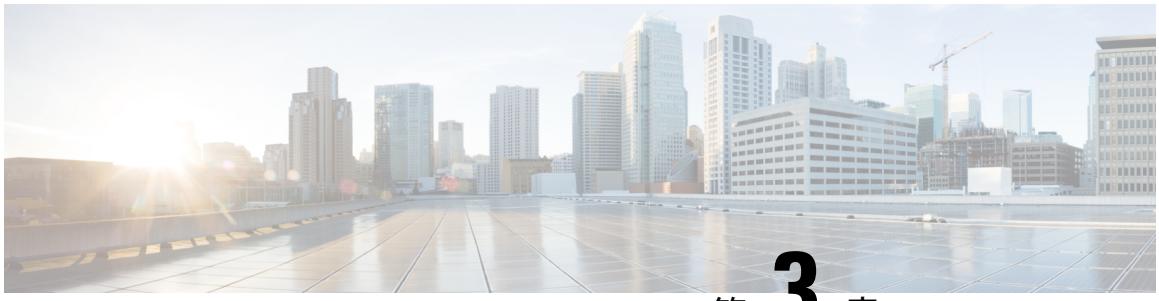
Cisco TrustSec VRF 対応 SGT の機能履歴

次の表に、このモジュールで説明する機能のリリースおよび関連情報を示します。

これらの機能は、特に明記されていない限り、導入されたリリース以降のすべてのリリースで使用できます。

リリース	機能	機能説明
Cisco IOS XE リリース 17.6.1	SGT および SGACL の IPv6 サポートの拡張	ホストから SGT へのマッピングとバインディング、およびサブネットから SGT へのバインディングを有効にします。
	SGACL 適用に対する IPv6 サポートの拡張	UDP ポート、TCP ポート、および上位層プロトコルタイプに基づいて、IPv6 トラフィックに SGACL を適用します。
	TrustSec FPGA モジュールでのポート範囲オプションのサポート	オプションは、スケーリングの問題に対処するためにポリシー要素でサポートされています。

リリース	機能	機能説明
Cisco IOS XE リリース 17.5.1	Cisco TrustSec VRF 対応 SGT	Cisco TrustSec VRF 対応 SGT 機能は、SGT SXP 接続を特定の VRF インスタンスにバインドします。



第 3 章

レイヤ 2 NAT の設定

L2 ネットワークアドレス変換（NAT）について

1 対 1（1：1）レイヤ 2 NAT は、固有のパブリック IP アドレスを既存のプライベート IP アドレス（エンド デバイス）に割り当てるサービスであり、エンド デバイスがプライベートとパブリック サブネット上で通信できるようになります。このサービスは、NAT 対応デバイスで設定され、エンドデバイスに物理的にプログラムされた IP アドレスのパブリックでの「エイリアス」です。これは、通常 NAT デバイスでテーブルとして表されます。

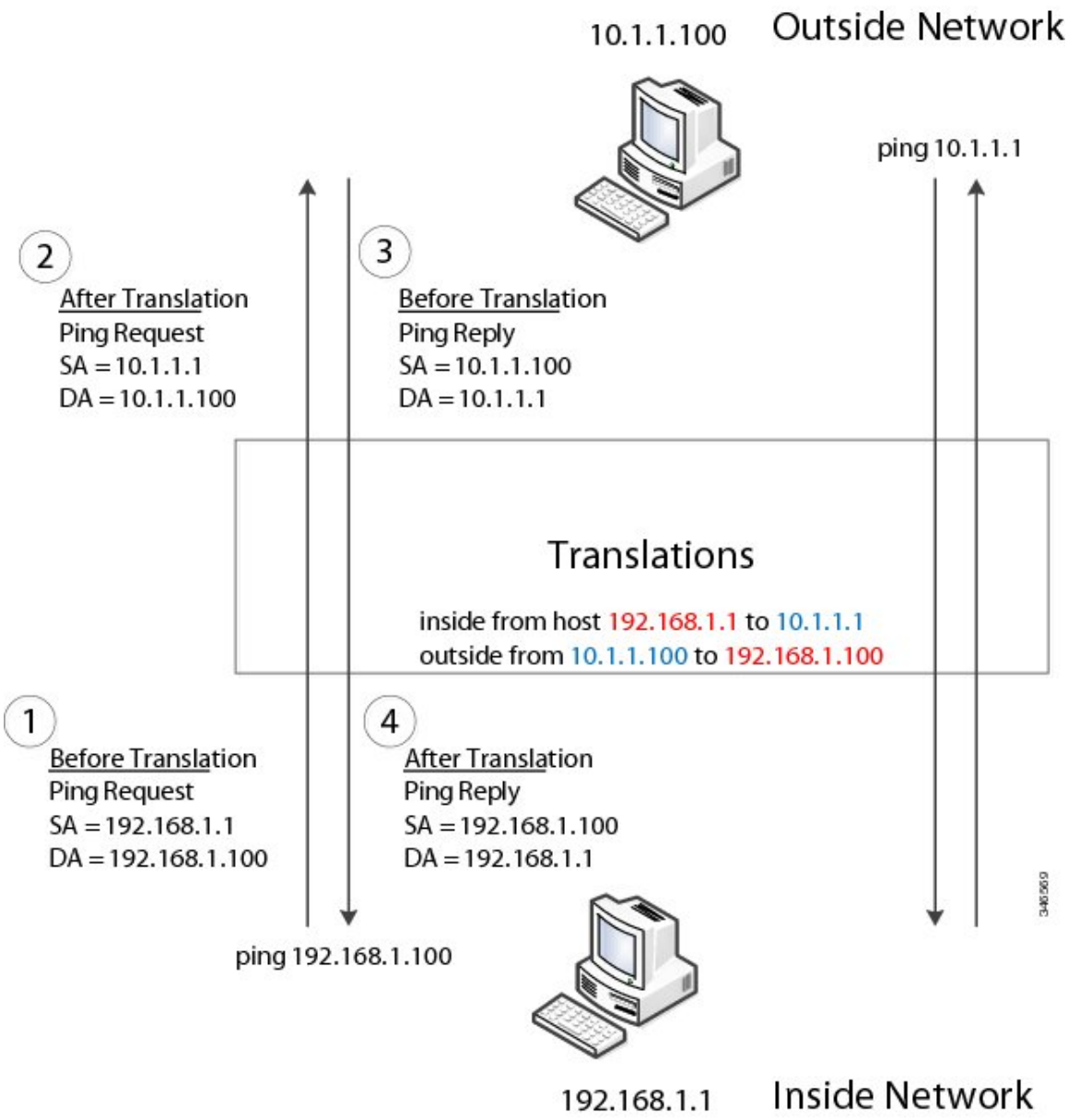
レイヤ 2 NAT には、プライベートからパブリックおよびパブリックからプライベートへサブネットの変換を定義できる 2 種類の変換テーブルがあります。レイヤ 2 NAT は、一貫した高レベルの（bump-in-the-wire）ワイヤスピードのパフォーマンスを提供するハードウェアベースの機能です。またこの機能は、拡張されたネットワーク セグメンテーション用の NAT 境界で複数の VLAN をサポートします。

次に、レイヤ 2 NAT で 192.168.1.x ネットワークのセンサーと 10.1.1.x ネットワークの通信制御装置間のアドレスを変換する例を示します。

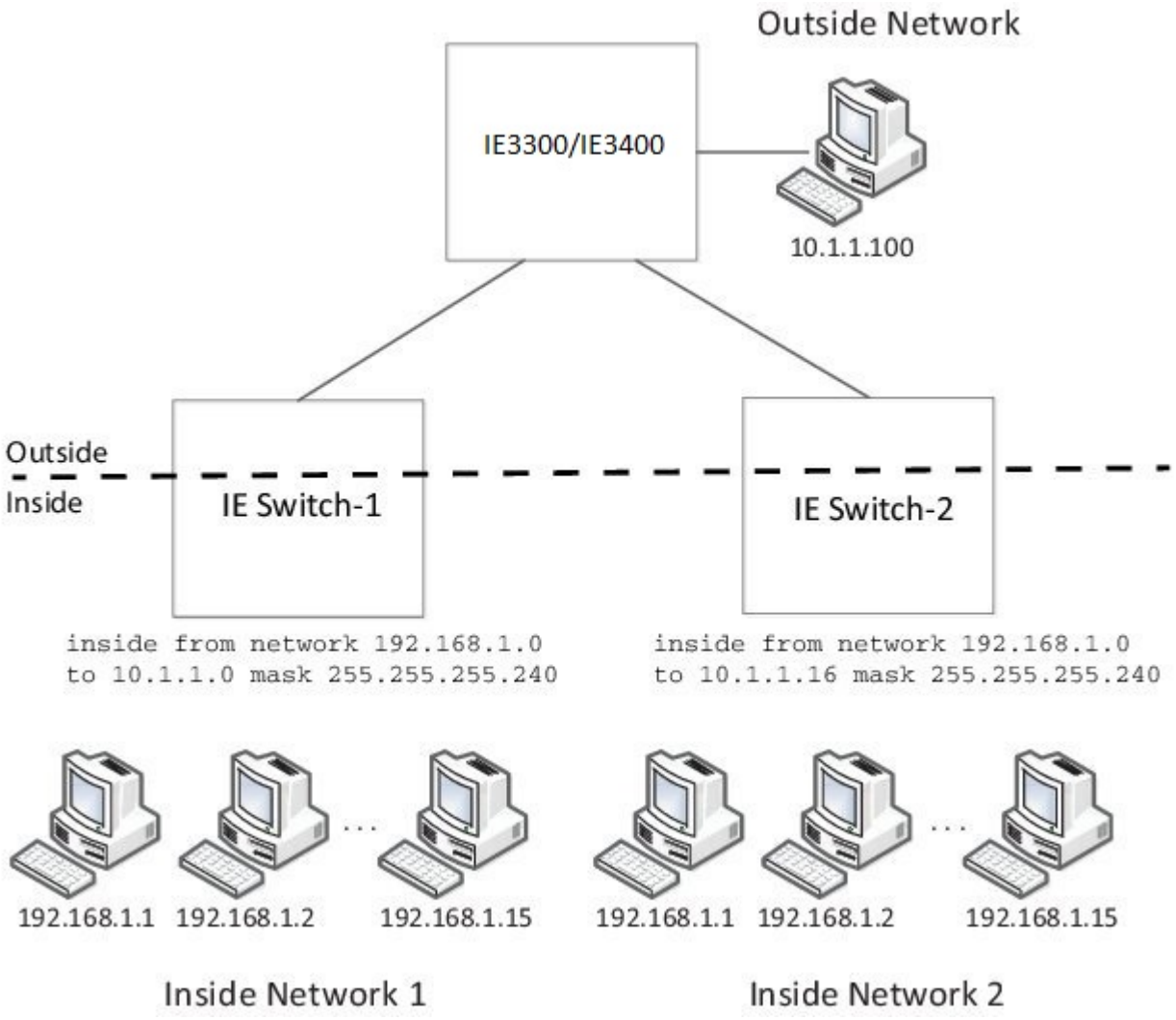
1. 192.168.1.x ネットワークは内部/内部 IP アドレス空間、10.1.1.x ネットワークは外部/外部 IP アドレス空間です。
2. 192.168.1.1 のセンサーが、「内部」アドレス 192.168.1.100 を使用して通信制御装置に ping 要求を送信します。
3. パケットが内部ネットワークから送信される前に、レイヤ 2 NAT は送信元アドレス（SA）を 10.1.1.1 へ、宛先アドレス（DA）を 10.1.1.100 へと変換します。

4. 通信制御装置は 10.1.1.1 へ ping 応答を送信します。
5. パケットが内部ネットワークで受信されると、レイヤ 2 NAT は送信元アドレスを 192.168.1.100 へ、宛先アドレスを 192.168.1.1 へ変換します。

図 *1*: ネットワーク間のアドレス変換

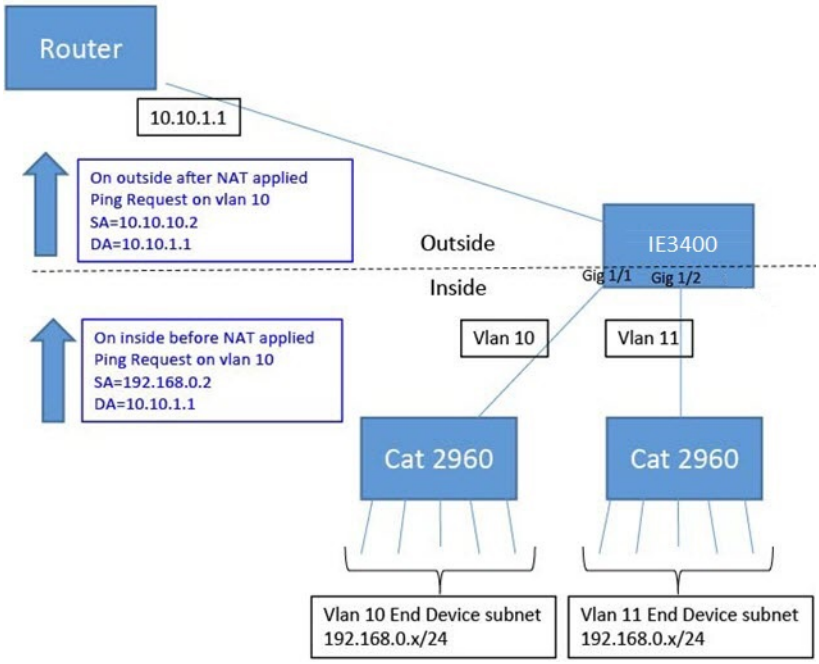


多数のノードに対して、サブネット内のすべてのデバイスの変換をまとめて有効にできます。この場合、内部ネットワーク 1 からのアドレスは 10.1.1.0/28 サブネットで外部アドレスに変換することができ、内部ネットワーク 2 からのアドレスは 10.1.1.16/28 サブネットで外部アドレスに変換することができます。各サブネットのアドレスはすべて 1 つのコマンドを使って変換できます。



次の図に、配布レベルでの IE 3400 NAT の設定を示します。この例では、IE 3400 は Catalyst 2960 スイッチを介してプライベートネットワーク内のデバイスに接続します。Catalyst スイッチは、アクセスレイヤで NAT を実行していません。IE 3400 では、2 つの異なるアクセススイッチ用の 2 つのインターフェイスで L2 NAT を実行しています。IE スイッチでは、128 個の L2 NAT インスタンスをサポートできます。この例では、128 個のうち 3 個のみ表示されています。サブネット全体を 1 つの L2 NAT インスタンスで設定できます。

図 *2 : IE 3400* での *NAT*



上の図に表示されている IE 3400 NAT の設定は次のとおりです。

```
Instance10:
 inside from network 192.168.0.0 to 10.10.10.0 mask 255.255.255.0
 outside from host 10.10.10.254 to 192.168.9.254 gateway
Instance11:
 inside from network 192.168.0.0 to 10.10.11.0 mask 255.255.255.0
 outside from host 10.10.11.254 to 192.168.9.254 gateway
.
.
.
Interface vlan 10
 ip address 10.10.10.254 mask 255.255.255.0
Interface vlan 11
 ip address 10.10.11.254 mask 255.255.255.0
Interface gig 1/1
 switchport access vlan 10
 l2nat instance10
Interface gig 1/2
 switchport access vlan 11
 l2nat instance11
```

前提条件

- IE 3300：L2NAT 機能は、アップリンクポート（Gig 1/1 および Gig 1/2）でのみサポートされており、両方の（Essential および Advantage）ライセンスで使用できます。
- IE 3400：L2NAT 機能は、アップリンクポート（Gig 1/1 および Gig 1/2）でのみサポートされており、両方の（Essential および Advantage）ライセンスで使用できます。

注意事項と制約事項

- IPv4 アドレスのみ変換できます。
- レイヤ 2 NAT はユニキャストトラフィックにのみ適用されます。未変換のユニキャストトラフィック、マルチキャストトラフィック、およびIGMPトラフィックを通過することができます。
- レイヤ 2 NAT は、1 対多および多対 1 の IP アドレスのマッピングをサポートしていません。
- レイヤ 2 NAT は、外部 IP アドレスと内部 IP アドレス間の 1 対 1 のマッピングをサポートしています。
- レイヤ 2 NAT ではパブリック IP アドレスを節約できません。
- レイヤ 2 NAT のホストの変換を設定する場合は、DHCP クライアントとして設定しないでください。
- ARP、ICMP などの特定のプロトコルは、レイヤ 2 NAT 越しに透過的に機能しませんが、これはデフォルトで「フィックスアップ」されます。「フィックスアップ」とは、プロトコルが機能するように IP パケットのペイロードに組み込まれた IP アドレスが変更されることを意味します。
- ダウンリンクポートには、VLAN、トランク、レイヤ 2 チャネルなどがあります。
- スイッチには、128 のレイヤ 2 NAT インスタンスを設定できます。
- レイヤ 2 NAT 設定では最大 128 の VLAN が利用できます。
- 管理インターフェイスはレイヤ2NAT機能の背後にあります。そのためこのインターフェイスはプライベート ネットワーク VLAN 上に置かないようにしてください。プライベート ネットワーク VLAN 上に存在する場合は、内部アドレスを割り当て、内部の変換を設定します。
- L2NATは外部アドレスと内部アドレスを分けるように設計されているため、同じサブネットのアドレスを外部アドレスと内部アドレスの両方に設定しないことを推奨します。
- NAT インスタンスの設定をサポートするインターフェイスは次のとおりです。
  - IE-3300 および IE3400：Gig 1/1 および Gig 1/2（アップリンク）

デフォルト設定

機能	デフォルト設定
一致しないトラフィックまたは変換するよう設定されていないトラフィックタイプのパケットの通過または破棄。	すべての一致しない、マルチキャストの IGMP パケットを破棄する。
プロトコル フィックスアップ	フィックスアップは、ARP および ICMP に対して有効になっています。

レイヤ 2 NAT の設定

アドレスの変換を指定するレイヤ 2 NAT インスタンスを設定する必要があります。その後、インターフェイスおよびVLANにこれらのインスタンスを接続します。一致しないトラフィック及び変換するよう設定されていないトラフィックタイプに対して、トラフィックの通過または破棄を選択できます。レイヤ 2 NAT インスタンスは、管理インターフェイス（CLI/SNMP/CIP/WebUI）から設定できます。送受信されたパケットに関する詳細な統計情報を確認できます（設定の確認 , on page 50を参照）。

レイヤ 2 NAT を設定するには、次の手順を実行します。詳細については、基本的な内部から外部への通信の例, on page 50および重複する IP アドレスの例, on page 52の例を参照してください。

SUMMARY STEPS

1. 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。
2. 新しいレイヤ 2 NAT インスタンスを作成します。
3. 内部アドレスを外部アドレスへ変換します。
4. 外部アドレスを内部アドレスへ変換します。
5. NAT 変換によって ICMP および IGMP の変換が修正されます。デフォルトでは、ARP と ICMP の両方のフィックスアップが有効になっているため、通常はデフォルトを変更しない限りこのコマンドは必要ありません。
6. （オプション）未変換のユニキャストトラフィックを通過します（デフォルトでは破棄されます）。
7. config-l2nat モードを終了します。
8. 指定したインターフェイス（IE 3400 のアップリンクポートのみ）のインターフェイス コンフィギュレーション モードにアクセスします。
9. VLAN または VLAN 範囲に指定されたレイヤ 2 NAT のインスタンスを適用します。このパラメータが欠落している場合、レイヤ 2 NAT インスタンスはネイティブ VLAN に適用されます。
10. 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。

DETAILED STEPS

ステップ 1 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。

configure terminal

ステップ 2 新しいレイヤ 2 NAT インスタンスを作成します。

l2nat instance *instance_name*

インスタンスを作成した後、そのインスタンスのサブモードを開始する場合もこのコマンドを使用します。

ステップ 3 内部アドレスを外部アドレスへ変換します。

inside from [*host* | *range* | *network*] *original ip* to *translated ip* [*mask*] *number* | *mask*

単一のホストアドレス、ホストアドレスの範囲、またはサブネット内のすべてのアドレスを変換できます。発信トラフィックの送信元アドレスと着信トラフィックの宛先アドレスを変換します。

ステップ 4 外部アドレスを内部アドレスへ変換します。

outside from [*host* | *range* | *network*] *original ip* to *translated ip* [*mask*] *number* | *mask*

単一のホストアドレス、ホストアドレスの範囲、またはサブネット内のすべてのアドレスを変換できます。発信トラフィックの宛先アドレスと着信トラフィックの送信元アドレスを変換します。

ステップ 5 NAT 変換によって ICMP および IGMP の変換が修正されます。デフォルトでは、ARP と ICMP の両方のフィックスアップが有効になっているため、通常はデフォルトを変更しない限りこのコマンドは必要ありません。

fixup arp | icmp | all

Note ICMP では、ICMP エラーメッセージに対するフィックスアップのみがサポートされます。

ステップ 6 （オプション）未変換のユニキャストトラフィックを通過します（デフォルトでは破棄されます）。

permit { **multicast** | **igmp** | **all** }

ステップ 7 config-l2nat モードを終了します。

exit

ステップ 8 指定したインターフェイス（IE 3400 のアップリンクポートのみ）のインターフェイス コンフィギュレーション モードにアクセスします。

interface *interface-id*

ステップ 9 VLAN または VLAN 範囲に指定されたレイヤ 2 NAT のインスタンスを適用します。このパラメータが欠落している場合、レイヤ 2 NAT インスタンスはネイティブ VLAN に適用されます。

l2nat *instance_name* [*vlan* | *vlan_range*]

ステップ 10 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。

end

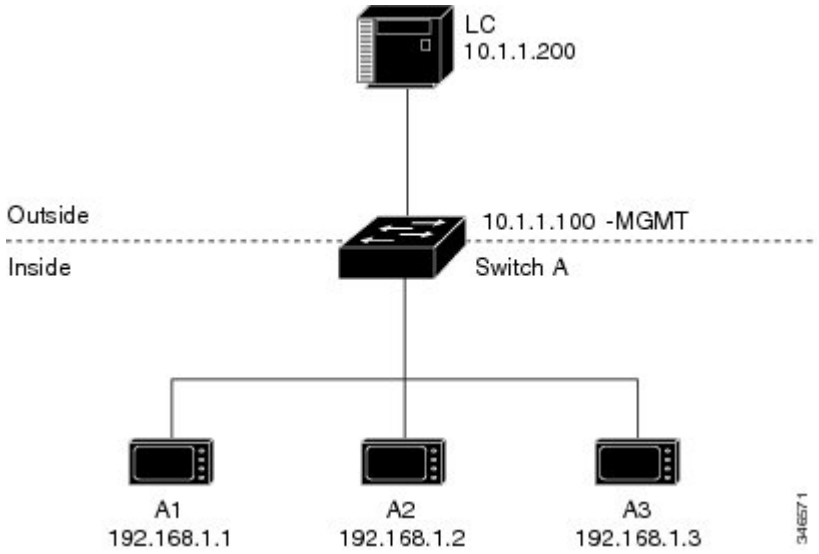
設定の確認

コマンド	目的
show l2nat instance	指定されたレイヤ 2 NAT インスタンスの設定の詳細を表示します。
show l2nat interface	1 つまたは複数のインターフェイスでのレイヤ 2 NAT インスタンスの設定の詳細を表示します。
show l2nat statistics	すべてのインターフェイスのレイヤ 2 NAT 統計情報を表示します。
show l2nat statistics interface	指定したインターフェイスのレイヤ 2 NAT 統計情報を表示します。
debug l2nat	設定が適用されたときにリアルタイムでのレイヤ 2 NAT 設定の詳細の表示を有効にします。

基本的な内部から外部への通信の例

ここでは、A1 はアップリンクポートに直接接続されたロジックコントローラ（LC）と通信する必要があります。レイヤ 2 NAT インスタンスは、外部ネットワーク（10.1.1.1）上での A1 のアドレスと内部ネットワーク（192.168.1.250）上での LC のアドレスを提供するように設定されています。

図 *3* : 基本的な内部から外部への通信



ここで次の通信が発生します。

1. A1 が「SA: 192.168.1.1DA: 192.168.1.250」という ARP 要求を送信します。
2. Cisco スイッチ A は「SA:10.1.1.1DA: 10.1.1.200」という ARP 要求をフィックスアップします。
3. LC は要求を受信し、10.1.1.1 の MAC アドレスを学習します。
4. LC が「SA: 10.1.1.200DA: 10.1.1.1」という応答を送信します。
5. Cisco スイッチ A は「SA: 192.168.1.250DA: 192.168.1.1」という ARP 応答をフィックスアップします。
6. A1 は 192.168.1.250 の MAC アドレスを学習し、通信を開始します。

（注） スイッチの管理インターフェイスは内部ネットワーク 192.168.1.x. とは別の VLAN に属している必要があります。

次の表は、このシナリオの設定作業を示しています。レイヤ2NATインスタンスが作成され、2つの変換エントリを追加し、インスタンスをインターフェイスに適用します。ARP フィックスアップはデフォルトで有効です。

表 *1* : 基本的な内部から外部への *Cisco* スイッチ *A* の設定例

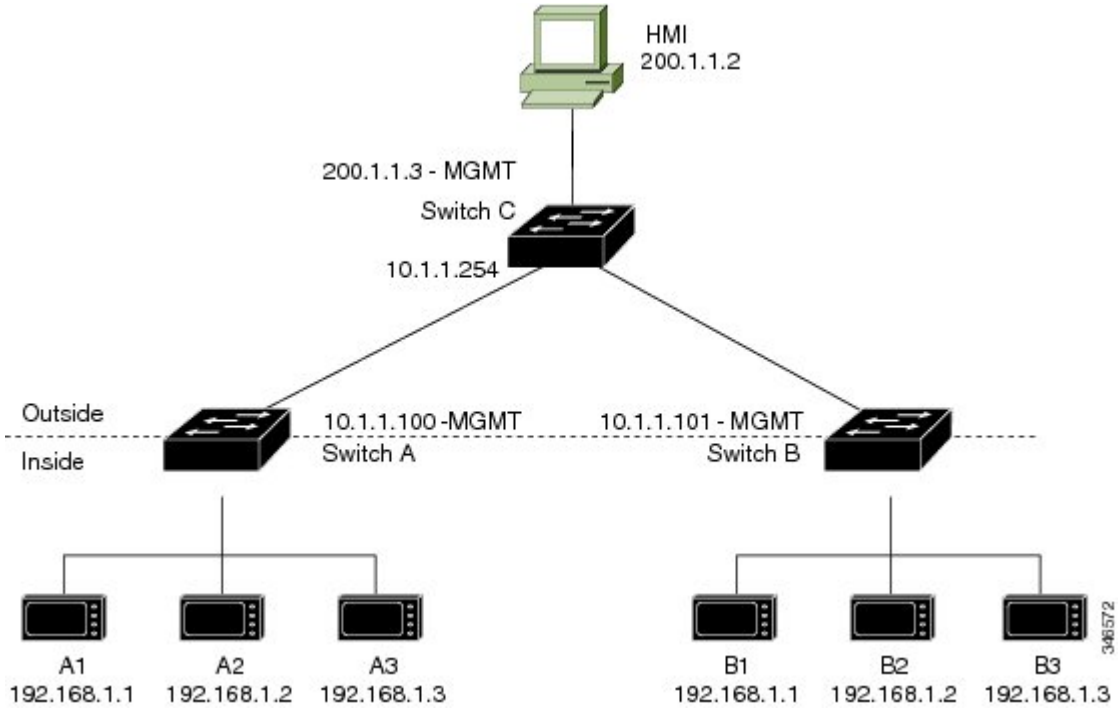
	コマンド	目的
1.	Switch# **configure**	グローバル コンフィギュレーション モードを開始します。
2.	Switch(config)# **l2nat instance A-LC**	A-LC という新しいレイヤ 2 NAT インスタンスを作成します。

	コマンド	目的
3.	Switch(config-l2nat)# **inside from host 192.168.1.1 to 10.1.1.1**	A1 の内部アドレスを外部アドレスへ変換します。
4.	Switch(config-l2nat)# **inside from host 192.168.1.2 to 10.1.1.2**	A2 の内部アドレスを外部アドレスへ変換します。
5.	Switch(config-l2nat)# **inside from host 192.168.1.3 to 10.1.1.3**	A3 の内部アドレスを外部アドレスへ変換します。
6.	Switch(config-l2nat)# **outside from host 10.1.1.200 to 192.168.1.250**	LC の外部アドレスを内部アドレスへ変換します。
7.	Switch(config-l2nat)# **exit**	config-l2nat モードを終了します。
8.	Switch(config)# **interface Gi1/1**	アップリンク ポートのインターフェイス コンフィギュレーション モードにアクセスします。
9.	Switch(config-if)# **l2nat A-LC**	このインターフェイスのネイティブ VLAN に、先ほどのレイヤ 2 NAT インスタンスを適用します。 （注） トランク上のタグ付きトラフィックの場合は、インターフェイスへインスタンスを適用するときに、次のように VLAN 番号を追加します。 l2nat *instance vlan*
10.	Switch# **end**	特権 EXEC モードに戻ります。

重複する IP アドレスの例

ここでは、2 台のマシンノードで 192.168.1.x 領域のアドレスが事前設定されています。レイヤ 2 NAT により、これらのアドレスが外部ネットワークの別のサブネット上で一意のアドレスに変換されます。また、マシン間の通信では、ノード A のマシンはノード B の領域で一意のアドレスを必要とし、ノード B のマシンはノード A の領域で一意のアドレスが必要です。

Figure 4: IP アドレスの重複



- スイッチ C は 192.168.1.x 領域でのアドレスが必要です。パケットがノード A またはノード B で受信されると、スイッチ C の 10.1.1.254 というアドレスが 192.168.1.254 に変換されます。パケットがノード A またはノード B から送信されると、スイッチ C の 192.168.1.254 というアドレスは 10.1.1.254 に変換されます。
- ノード A とノード B のマシンは 10.1.1.x 領域で一意のアドレスが必要です。設定の容易さと使いやすさを実現するために、10.1.1.x 領域は 10.1.1.0、10.1.1.16、10.1.1.32 などのサブネットに分割されます。各サブネットは異なるノードに使用できます。この例では、10.1.1.16 はノード A に使用され、10.1.1.32 はノード B に使用されます。
- ノード A とノード B のマシンはデータを交換するための一意のアドレスが必要です。使用可能なアドレスはサブネットに分割されます。便宜上、ノード A のマシンの 10.1.1.16 サブネットアドレスは、ノード B の 192.168.1.16 サブネットアドレスに変換され、ノード B のマシンの 10.1.1.32 サブネットアドレスはノード A の 192.168.1.32 アドレスに変換されます。
- マシンは各ネットワークで一意のアドレスを持ちます。

Table 2: IP アドレスの変換

ノード	ノード A のアドレス	外部ネットワークのアドレス	ノード B のアドレス
スイッチ A のネットワークアドレス	192.168.1.0	10.1.1.16	192.168.1.16
A1	192.168.1.1	10.1.1.17	192.168.1.17

ノード	ノード A のアドレス	外部ネットワークのアドレス	ノード B のアドレス
A2	192.168.1.2	10.1.1.18	192.168.1.18
A3	192.168.1.3	10.1.1.19	192.168.1.19
Cisco スイッチ B のネットワークアドレス	192.168.1.32	10.1.1.32	192.168.1.0
B1	192.168.1.33	10.1.1.33	192.168.1.1
B2	192.168.1.34	10.1.1.34	192.168.1.2
B3	192.168.1.35	10.1.1.35	192.168.1.3
スイッチ C	192.168.1.254	10.1.1.254	192.168.1.254

Table 3: アドレスが重複するスイッチ A の設定例, on page 54に、スイッチ A の設定作業を示します。スイッチ B の設定作業については、Table 4: サブネットのスイッチ B の設定例, on page 55に示します。

Note この例は、IE 2000 スイッチに基づいています。IE3x00 および ESS3300 スイッチでは、インターフェイスの番号が異なる場合があります。

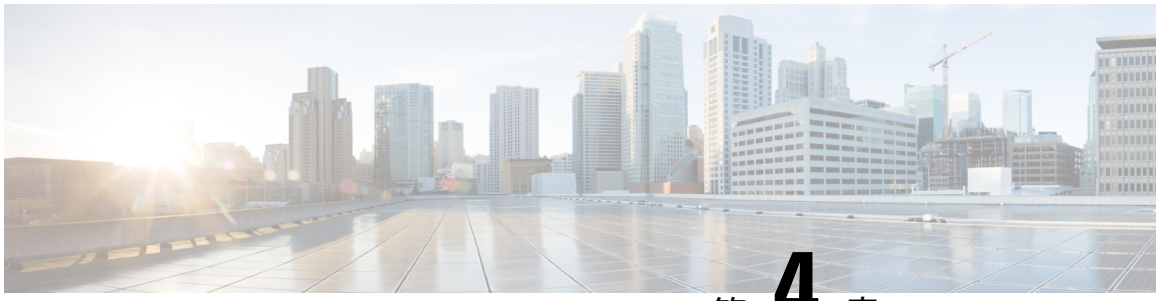
Table 3: アドレスが重複するスイッチ *A* の設定例

	コマンド	目的
1	Switch# **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。
2	Switch(config)# **l2nat instance A-Subnet**	A-Subnet という新しいレイヤ 2 NAT インスタンスを作成します。
3	Switch(config-l2nat)# **inside from network 192.168.1.0 to 10.1.1.16 mask 255.255.255.240**	ノード A のマシンの内部アドレスを 10.1.1.16 255.255.255.240 サブネットのアドレスへ変換します。
4	Switch(config-l2nat)# **outside from host 10.1.1.254 to 192.168.1.254**	スイッチ C の外部アドレスを内部アドレスへ変換します。
5	Switch(config-l2nat)# **outside from network 10.1.1.32 to 192.168.1.32 255.255.255.240**	ノード B のマシンの外部アドレスを内部アドレスへ変換します。
6	Switch(config-l2nat)# **exit**	config-l2nat モードを終了します。
7	Switch(config)# **interface Gi1/1**	アップリンク ポートのインターフェイス コンフィギュレーション モードにアクセスします。

	コマンド	目的
8.	Switch(config-if)# **l2nat A-Subnet**	このインターフェイスのネイティブ VLAN に、先ほどのレイヤ 2 NAT インスタンスを適用します。 **Note** トランク上のタグ付きトラフィックの場合は、インターフェイスへインスタンスを適用するときに、次のように VLAN 番号を追加します。 l2nat *instance vlan*
9.	Switch# **end**	特権 EXEC モードに戻ります。

Table 4: サブネットのスイッチ *B* の設定例

	コマンド	目的
1.	Switch# **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。
2.	Switch(config)# **l2nat instance B-Subnet**	B-Subnet という新しいレイヤ 2 NAT インスタンスを作成します。
3.	Switch(config-l2nat)# **inside from network 192.168.1.0 to 10.1.1.32 255.255.255.240**	ノード B のマシンの内部アドレスを 10.1.1.32 255.255.255.240 サブネットのアドレスへ変換します。
4.	Switch(config-l2nat)# **outside from host 10.1.1.254 to 192.168.1.254**	スイッチ C の外部アドレスを内部アドレスへ変換します。
5.	Switch(config-l2nat)# **outside from network 10.1.1.16 to 192.168.1.16 255.255.255.240**	ノード A のマシンの外部アドレスを内部アドレスへ変換します。
6.	Switch(config-l2nat)# **exit**	config-l2nat モードを終了します。
7.	Switch(config)# **interface Gi1/1**	アップリンク ポートのインターフェイス コンフィギュレーション モードにアクセスします。
8.	Switch(config-if)# **l2nat name1**	このインターフェイスのネイティブ VLAN に、先ほどのレイヤ 2 NAT インスタンスを適用します。 **Note** トランク上のタグ付きトラフィックの場合は、インターフェイスへインスタンスを適用するときに、次のように VLAN 番号を追加します。 l2nat *instance vlan*
9.	Switch# **show l2nat instance name1**	指定されたレイヤ 2 NAT インスタンスの設定の詳細を表示します。
10.	Switch# **show l2nat statistics**	レイヤ 2 NAT の統計情報を表示します。
11.	Switch# **end**	特権 EXEC モードに戻ります。



第 4 章

Cisco Secure Cloud Analytics コネクタの設定

Cisco Connector for Secure Cloud Analytics の設定

Cisco Secure Cloud Analytics（旧称 Stealthwatch Cloud）は、悪意のある各種アクティビティをリアルタイムで特定するために必要な、実用的なセキュリティインテリジェンスと可視性を提供します。セキュリティインシデントが壊滅的な侵害になる前に迅速に対応できます。このガイドでは、シスコ産業用イーサネットスイッチでの IOS-XE での Cisco Cloud Connector の設定手順について説明します。

（注） Cisco Secure Cloud Analytics（Stealthwatch Cloud）または Cisco Secure Network Analytics（Stealthwatch）の詳細については、次の URL を参照してください。https://www.cisco.com/c/en/us/products/security/stealthwatch/index.html

制限事項と制約事項

- 事前に定義された一連のフィールドのみを収集できます。対象のフィールドには、送信元 IP、送信元ポート、宛先 IP、宛先ポートおよびプロトコルの 9 タプルフローデータと、フロー開始、フロー終了、パケット数、およびバイト数が含まれます。
- 必須フィールドは、CLI の制限では適用されません。レコードにすべての必須フィールドがなく、9 タプルデータを収集できない場合、そのフローは破棄されます。
- Cisco Secure Cloud Analytics 用の StealthWatch コネクタは、スイッチのルーティング機能を使用して、クラウドサーバにパケットを送信します。追加のチェックは行われません。適切なルートが存在することを前提としています。

- モニタアプリケーションの観点から Flexible NetFlow 固有のモニタアプリケーションの制限は、Cisco Secure Cloud Analytics にも当てはまります。例：SVI なし、VLAN なし、送信モニタなし。
- クラウドエクスポータを他のエクスポータと一緒に使用することはできません。
- アップロードされたファイルの命名規則には、すべてのファイルを一意に識別し、ファイルの上書きを防ぐためのランダムな文字列が含まれています。例：https://sensor.ext.obsrvbl.com/sign/ios-xe-17-2/2019/7/5/00:00:00/hostname-random_suffix.csv.gz。1 分ごとに集約されてアップロードされます。

始める前に

Cisco Secure Cloud Analytics コネクタは、IE3300、IE3400、IE3400H スイッチでのみサポートされます。

- Network Advantage および Cisco DNA Advantage ライセンス

手順の概要

1. stealthwatch-cloud-monitor service-key <you service key> hostname my_sensor
2. フローレコード SWCRec
3. フローエクスポータ SWCExp
4. インターフェイス gi1/0/3

手順の詳細

	コマンドまたはアクション	目的

ステップ 1	stealthwatch-cloud-monitor service-key <you service key> hostname my_sensor
	例：
	`stealthwatch-cloud-monitor`
	`  service-key <you service key>`
	`  hostname my_sensor`
	`  url https://sensor.ext.obsrvbl.com`
	`openssl s_client -showcerts -connect`
	`https://sensor.ext.obsrvbl.com:443`
	`openssl s_client -showcerts -connect`
	`s3.ap-southeast-2.amazonaws.com:443`
	例：
	`openssl s_client -showcerts -connect`
	`https://sensor.ext.obsrvbl.com:443`
	`openssl s_client -showcerts -connect`
	`s3.ap-southeast-2.amazonaws.com:443`
		URL に基づいて有効なルート CA をインストールしてください。以下の CLI を使用して、URL に従ってルート CA を把握してください
		センサーの登録に使用されるサービスキーとホスト名を設定します。ホスト名を指定しない場合は、ボックスのシリアル番号が登録に使用されます。

ステップ 2	フローレコード SWCRec
	例：
	`flow record SWCRec`
	`match ipv4 source address`
		Cisco Secure Cloud Analytics レコードのデータを収集するためのフローレコードのフィールドを設定します。

	コマンドまたはアクション	目的
	match ipv4 destination address match transport source-port match transport destination-port match ipv4 protocol collect counter bytes long collect counter packets long collect timestamp sys first collect timestamp sys last	
ステップ 3	フローエクスポータ SWCExp **例：** flow exporter SWCExp destination stealthwatch-cloud flow monitor SWCMon flow record SWCRec flow exporter SWCExp	Cisco Secure Cloud Analytics エクスポータを設定し、フローモニタに接続して、Secure Cloud へのエクスポートを開始します。
ステップ 4	インターフェイス gi1/0/3 **例：** Interface gi1/0/3 ip flow monitor SWCMon input	フローをモニタするインターフェイスを特定し、Cisco Secure Cloud Analytics エクスポータがあるモニタをそのインターフェイスに接続します。

次のタスク

Cisco Secure Cloud Analytics の詳細情報については、該当するコンフィギュレーション ガイドを参照してください。https://www.cisco.com/c/en/us/support/security/stealthwatch-cloud/products-installation-and-configuration-guides-list.html

トラブルシューティング

- デバッグログは「debug Stealthwatch」CLI を使用して有効にできます。

```
switch#debug stealthwatch-cloud ?
  all          All debugs for SWC
  cert         Certificate Validation
  error        errors
  event        Events
  file-events  File notifications
```

- プラットフォームレベルのデバッグでは、「debug platform software swc」CLI を使用できます。

```
switch#debug platform software swc ?
  all         all
  errors      Stealthwatch Cloud errors
  events      Stealthwatch Cloud events
  pkt-events  Stealthwatch Cloud data collection events
```

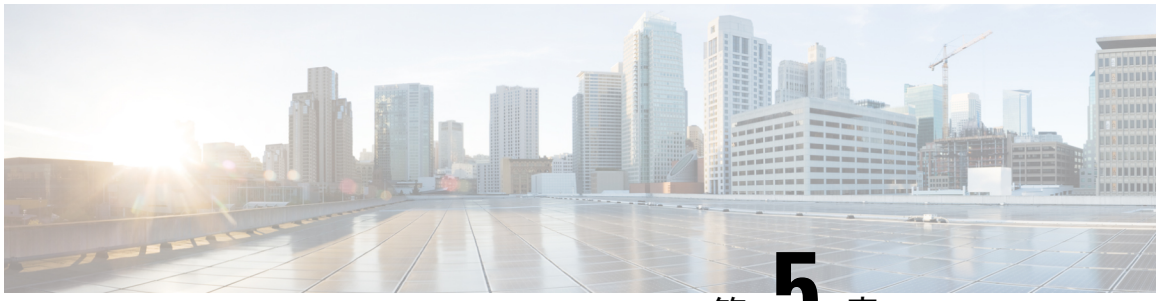
コマンドの表示

- **Switch-1# show stealthwatch-cloud detail**

```
=======================================
Stealthwatch Cloud Parameters
=======================================
   Service Key  : x8SS2q7e4twpcNWT35AsL6i6xHd24iXJvICo3N4sGx1U1pCqqs
   Sensor Name  : petra
   URL          : https://sensor.anz-prod.obsrvbl.com
=======================================
Stealthwatch Cloud Sensor Info
=======================================
   Sensor Status    : Registered
   Last heartbeat   : 2020-05-08T12:11:50
```

- **Switch-1# show platform software swc stats**

```
=========================
SWC Upload Statistics:
=========================
1 : Last file uploaded          :  202005081212_ufihi2
2 : Time of upload              :  202005081213 UTC
3 : Current file uploading      :
4 : Files queued for upload     :
5 : Number of files queued      :  0
6 : Last failed upload          :
7 : Files failed to upload      :  0
8 : Files successfully uploaded :  416
=========================
SWC File Creation Statistics:
=========================
9 : Last file created           :  202005081212_ufihi2
10: Time of creation            :  202005081212 UTC
=========================
SWC Flow Statistics:
=========================
11: Number of flows in prev file:  1
12: Number of flows in curr file:  0
13: Invalid dropped flows       :  0
=========================
SWC Flags:
=========================
14: Is Registered               :  Registered
15: File Delete         :  Enabled
16: Exporter            :  Enabled
```



第 5 章

Cisco Umbrella 統合

Cisco Umbrella 統合の前提条件

- Cisco Umbrella サブスクリプション ライセンスが利用可能である必要があります。https://umbrella.cisco.com/products/umbrella-enterprise-security-packages に移動し、[Request a quote] をクリックしてライセンスを取得します。
- デバイスはデフォルトのドメインネームシステム（DNS）サーバゲートウェイとして設定する必要があり、ドメインネームサーバのトラフィックはシスコデバイスを通過する必要があります。
- Umbrella サーバへのデバイス登録に使用する通信は HTTPS 経由です。HTTPS 通信を行うには、デバイスにルート証明書がインストールされている必要があります。次のリンクを使用して証明書をダウンロードできます。https://www.digicert.com/CACerts/DigiCertSHA2SecureServerCA.crt
- シスコ産業用イーサネットスイッチは、Cisco IOS XE リリース 17.2.1 以降のソフトウェアイメージを実行します。
- Cisco Umbrella を有効にするには、シスコ産業用イーサネットスイッチに Cisco DNA Advantage 以上のライセンスが必要です。

次のネットワーク要件を満たす必要があります。

- デバイスをデフォルトの DNS サーバゲートウェイとして設定し、ドメインネームサーバ（DNS）トラフィックがシスコ産業用イーサネットスイッチを通過するようにします。
- Cisco Umbrella サーバへのデバイス登録に使用する通信は HTTPS 経由です。HTTPS 通信を行うには、ルータにルート証明書がインストールされている必要があります。この証明書をペーストする代わりに、次のリンクから証明書を直接ダウンロードすることができます。

  https://www.digicert.com/CACerts/DigiCertSHA2SecureServerCA.crt
- 最初の登録の場合、「umbrella out」として設定されたインターフェイスは、最初の登録を完了するために、ポート 443 を介して api.opendns.com にアクセスできる必要があります。

Cisco Umbrella 統合の制限

- Cisco Umbrella 統合は、次のシナリオでは機能しません。
  - アプリケーションまたはホストが、DNS の代わりに IP アドレスを使用してドメイン名をクエリしている場合。
  - クライアントが Web プロキシに接続されていて、サーバアドレスを解決するための DNS クエリを送信しない場合。
  - DNS クエリがシスコスイッチデバイスによって生成される場合。
  - DNS クエリが TCP 経由で送信される場合。
  - DNS クエリに、アドレスマッピングとテキスト以外のレコードタイプがある場合。
- DNSv6 クエリはサポートされていません。
- DNS64 および DNS46 拡張はサポートされていません。
- 拡張 DNS は、ホストの IPv4 アドレスのみを伝達し、IPv6 アドレスは伝達しません。
- ポートチャネルでの Cisco Umbrella 設定はサポートされていません
- Cisco Umbrella は、10G アップリンクポートを送信専用として使用するように設定できます。
- Cisco Umbrella インターフェイスを経由する DNS トラフィックの DSCP マーキングは行われません。これは、Cisco Umbrella インターフェイス上のすべてのパントされたトラフィックに適用されます。
- Cisco Umbrella インターフェイスの場合、すべての送信 ACL ルールは DNS トラフィックに影響を及ぼしません。これは、DNS の CPU 処理されるトラフィックに適用されます。
- DNS パケットのフラグメンテーションはサポートされていません。
- QinQ およびセキュリティグループタグ（SGT）パケットはサポートされていません。

- Cisco Umbrella の統合ポリシーによって DNS クエリがブロックされると、クライアントは Cisco Umbrella ブロックページにリダイレクトされます。これらのブロックページは、HTTPS サーバによって提供され、IP アドレス範囲は Cisco Umbrella ポータルによって定義されます。
- ユーザ認証とアイデンティティは、現在サポートされていません。
- Cisco Umbrella Connector は、悪意のあるトラフィックに関する既知の IP アドレスのリストを保持します。Cisco Umbrella ローミングクライアントは、これらのアドレスが宛先のパケットを検出すると、各アドレスを Cisco Umbrella クラウドに転送して、さらに検査します。
- 現在、直接クラウドアクセスはサポートされていません。
- 更新されたリゾルバ IP は有効になりません。DNS トラフィックは、ユーザが設定したリゾルバ IP に関係なく、Cisco Umbrella クラウドにリダイレクトされます。
- ネットワークアドレス変換（NAT）は、Cisco Umbrella が有効になっているインターフェイスではサポートされません。

Cisco Umbrella 統合に関する情報

ここでは、Cisco Umbrella 統合機能の詳細を説明します。

Cisco Umbrella 統合のメリット

Cisco Umbrella 統合は、DNS レベルでのセキュリティとポリシーの適用を提供します。これにより、管理者は DNS トラフィックを分割して、DNS トラフィックの一部をエンタープライズネットワーク内にある特定の DNS サーバに直接送信することができます。これにより、管理者は Cisco Umbrella 統合をバイパスできます。

Cisco Umbrella 統合を使用したクラウドベースのセキュリティサービス

Cisco Umbrella 統合機能は、Cisco デバイスを介して DNS サーバに送信される DNS クエリを検査する、クラウドベースのセキュリティサービスを提供します。ホストがトラフィックを開始し、DNS クエリを送信すると、デバイスの Cisco Umbrella コネクタは DNS クエリを横取りして検査します。Umbrella コネクタは、DNS トラフィックを横取りして、セキュリティ検査およびポリシー適用のために Cisco Umbrella クラウドへのリダイレクトを行うシスコ デバイス内のコンポーネントです。Umbrella クラウドは、Umbrella コネクタから受信したクエリを検査するクラウドベースのセキュリティサービスであり、完全修飾ドメイン名（FQDN）に基づいて、コンテンツプロバイダーの IP アドレスを応答に含めるかどうかを決定します。

ローカルドメインへの DNS クエリの場合、DNS パケットを変更せずに企業ネットワーク内の DNS サーバにクエリが転送されます。Cisco Umbrella リゾルバは、外部ドメインから送信され

た DNS クエリを検査します。デバイス ID 情報、組織 ID、クライアント IP アドレスを含む拡張 DNS レコードがクエリに追加され、Cisco Umbrella リゾルバに送信されます。Umbrella クラウドは、このすべての情報に基づいて、DNS クエリにさまざまなポリシーを適用します。

Umbrella 統合クラウドは、ポータルで設定されたポリシーと DNS FQDN のレピュテーションに基づいて、次のいずれかのアクションを実行します。

- ブラックリストのアクション：FQDN が悪意のあるものであるか、カスタマイズされたエンタープライズセキュリティポリシーによってブロックされていると判明した場合、Cisco Umbrella クラウドのブロックランディングページの IP アドレスが DNS 応答で返されます。
- ホワイトリストのアクション：FQDN が悪意のないものであると判明した場合、コンテンツプロバイダーの IP アドレスが DNS 応答で返されます。
- グレーリストのアクション：FQDN が疑わしいと判明した場合、インテリジェントプロキシのユニキャスト IP アドレスが DNS 応答で返されます。

DNS 応答を受信すると、デバイスは応答をホストに転送します。ホストは応答から IP アドレスを抽出し、HTTP または HTTPS 要求をこの IP アドレスに送信します。

Cisco Umbrella クラウドによるトラフィックの処理

Cisco Umbrella 統合機能を使用すると、HTTP および HTTPS クライアント要求は次のように処理されます。

- DNS クエリの FQDN が悪意のあるものである場合（ブラックリストに登録されたドメインに含まれる場合）、Cisco Umbrella クラウドは DNS 応答でブロック時用ランディングページの IP アドレスを返します。HTTP クライアントがこの IP アドレスに要求を送信すると、Umbrella クラウドは、要求されたページがブロックされたことをユーザに通知するページと、ブロックの理由を表示します。
- DNS クエリの FQDN が悪意のないものである場合（ホワイトリストに登録されたドメインに含まれる場合）、Cisco Umbrella クラウドはコンテンツプロバイダーの IP アドレスを返します。HTTP クライアントはこの IP アドレスに要求を送信し、要求されたコンテンツを取得します。
- DNS クエリの FQDN がグレーリストのドメインに該当する場合、Umbrella DNS リゾルバは DNS 応答でインテリジェントプロキシのユニキャスト IP アドレスを返します。ホストからグレードメインへのすべての HTTP トラフィックは、インテリジェントプロキシを介してプロキシされ、Uniform Resource Locator（URL）フィルタリングが実行されます。

（注） インテリジェントプロキシのユニキャスト IP アドレスを使用する場合の潜在的な制限の 1 つは、クライアントがインテリジェントプロキシのユニキャスト IP アドレスにトラフィックを送信しようとしたときにデータセンターがダウンする可能性です。このシナリオでは、クライアントはグレーリストのドメインに該当するドメインの DNS 解決を完了し、クライアントの HTTP または HTTPS トラフィックは、取得されたインテリジェントプロキシのユニキャスト IP アドレスのいずれかに送信されます。そのデータセンターがダウンしている場合、クライアントはそれを知る方法がありません。

Umbrella コネクタは、HTTP および HTTPS トラフィックに対して動作しません。コネクタは、Web トラフィックをリダイレクトしたり、HTTP または HTTPS パケットを変更したりしません。

DNS パケット暗号化

Cisco デバイスから Cisco Umbrella 統合サーバに送信される DNS パケットは、パケット内の拡張 DNS 情報にユーザ ID、内部ネットワーク IP アドレスなどの情報が含まれている場合、暗号化する必要があります。DNS 応答が DNS サーバから戻されると、デバイスはパケットを復号化してからホストに転送します。

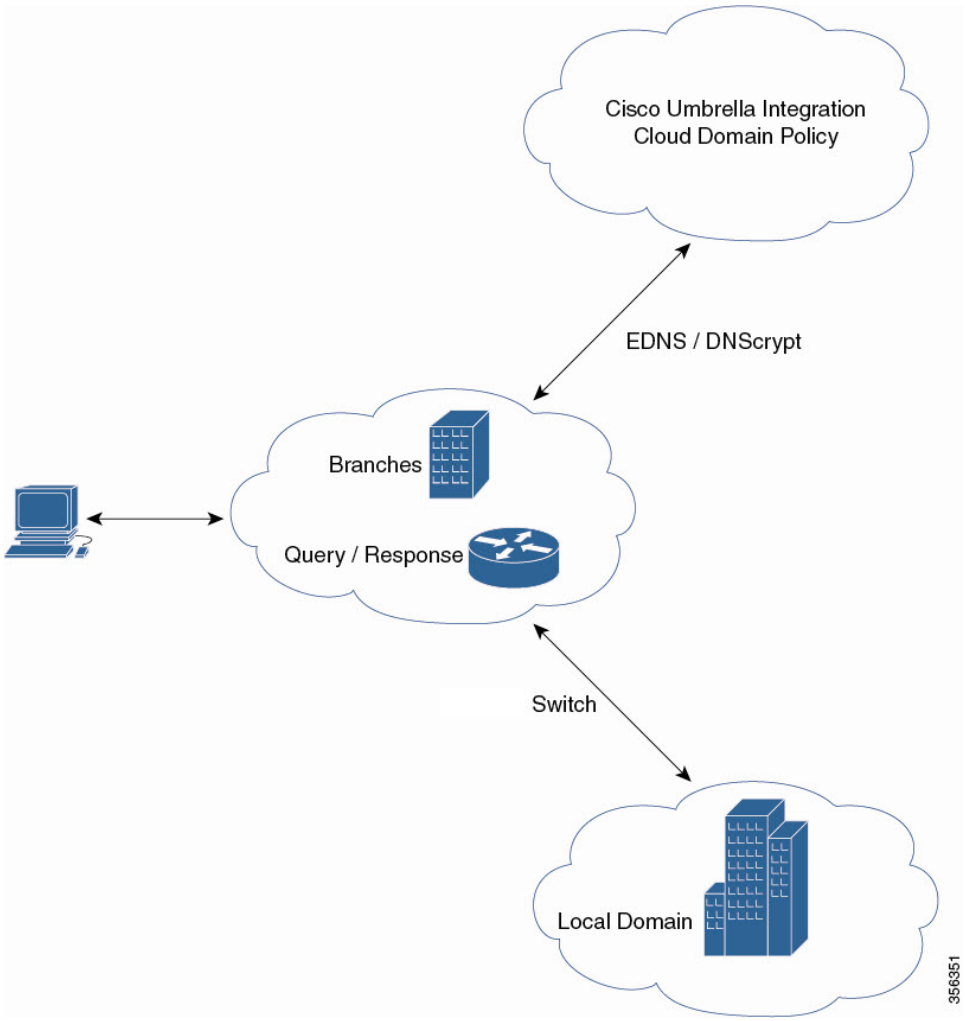
（注）
- DNS パケットは、DNScrypt 機能が Cisco デバイスで有効化されている場合にのみ暗号化できます。
- 統計情報を追跡するために、クライアントの IP アドレスが Umbrella クラウドにエクスポートされます。IP は暗号化されずに送信されるため、DNScrypt を無効にしないことを推奨します。

Cisco デバイスは次の Anycast 再帰型 Cisco Umbrella 統合サーバを使用します。

- 208.67.222.222
- 208.67.220.220
- 2620:119:53::53
- 2620:119:35::35

次の図に、Cisco Umbrella 統合のトポロジを示します。

図 *5 : Cisco Umbrella* 統合のトポロジ



DNSCrypt と公開キー

次のサブセクションでは、DNScrypt と公開キーについて詳しく説明します。

DNSCrypt

DNSCrypt は、Cisco デバイスと Cisco Umbrella 統合機能間の通信を認証する暗号化プロトコルです。**parameter-map type umbrella** が設定され、WAN インターフェイスで **umbrella out** コマンドが有効化されると、DNSCrypt がトリガーされ、証明書のダウンロード、検証、解析が行われます。次に、DNS クエリの暗号化に使用される共有秘密鍵のネゴシエーションが行われます。一時間おきにこの証明書が自動的にダウンロードされ、アップグレードのために検証され、その都度新しい共有秘密キーがネゴシエートされ、DNS クエリが暗号化されます。

DNSCrypt を使用する場合は、DNS 要求パケットサイズが 512 バイトよりも大きくなります。これらのパケットが中間デバイスを通過できることを確認します。そうしないと、応答が目的の受信者に到達しない可能性があります。

公開キー

公開キーは、Umbrella クラウドから DNSCrypt 証明書をダウンロードするために使用されます。この値は、B735:1140:206F:225D:3E2B:D822:D7FD:691E:A1C3:3CC8:D666:8D0C:BE04:BFAB:CA43:FB79（Cisco Umbrella Integration Anycast サーバの公開キー）に事前に設定されています。公開キーに変更があり、**public-key** コマンドを変更する場合、デフォルト値に戻すときは変更されたコマンドを削除する必要があります。

⚠

注意 この値を変更すると、DNSCrypt 証明書のダウンロードは失敗することがあります。

parameter-map type umbrella global コマンドは、Umbrella モードでパラメータマップタイプを設定します。このコマンドを使用してデバイスを設定すると、DNSCrypt と公開キーの値が自動入力されます。

ラボで特定のテストを実行するときは、**parameter-map type umbrella global** パラメータのみを変更することをお勧めします。これらのパラメータを変更すると、デバイスの正常な機能に影響が及ぶことがあります。

Cisco Umbrella 統合の設定方法

ここでは、Cisco Umbrella 統合を構成するさまざまな作業について説明します。

Umbrella Connector の設定

Before you begin

Cisco Umbrella 登録サーバからアプリケーション プログラミング インターフェイス（API）トークンを取得します。

Cisco Umbrella 登録サーバとの間で HTTPS 接続を確立するために、ルート証明書を取得します。グローバル コンフィギュレーション モードで **crypto pki trustpool import terminal** コマンドを使用して、DigiCert のルート証明書をデバイスにインポートします。

証明書をインポートする方法は 2 つあります。

1. URL からインポートする
2. 端末で直接インポートする

URL からインポートするには、コマンドを発行し、産業用イーサネットスイッチが証明書を取得できるようにします。

crypto pki trustpool import url http://www.cisco.com/security/pki/trs/ios.p7b

端末からインポートするには、次の手順を実行します。

DigiCert のルート証明書は次のとおりです。

```
-----BEGIN CERTIFICATE-----
MIIElDCCA3ygAwIBAgIQAf2j627KdciIQ4tyS8+8kTANBgkqhkiG9w0BAQsFADBh
MQswCQYDVQQGEwJVUzEVMBMGA1UEChMMRGlnaUNlcnQgSW5jMRkwFwYDVQQLExB3
d3cuZGlnaWNlcnQuY29tMSAwHgYDVQQDExdEaWdpQ2VydCBHbG9iYWwgUm9vdCBD
QTAeFw0xMzAzMDgxMjAwMDBaFw0yMzAzMDgxMjAwMDBaME0xCzAJBgNVBAYTAlVT
MRUwEwYDVQQKEwxEaWdpQ2VydCBJbmMxJzAlBgNVBAMTHkRpZ2lDZXJ0IFNIQTIg
U2VjdXJlIFNlcnZlciBDQTCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEB
ANyuWJBNwcQwFZA1W248ghX1LFy949v/cUP6ZCWA1O4Yok3wZtAKc24RmDYXZK83
nf36QYSvx6+M/hpzTc8zl5CilodTgyu5pnVILR1WN3vaMTIa16yrBvSqXUu3R0bd
KpPDkC55gIDvEwRqFDu1m5K+wgdlTvza/P96rtxcflUxDOg5B6TXvi/TC2rSsd9f
/ld0Uzs1gN2ujkSYs58O09rg1/RrKatEp0tYhG2SS4HD2nOLEpdIkARFdRrdNzGX
kujNVA075ME/OV4uuPNcfhCOhkEAjUVmR7ChZc6gqikJTvOX6+guqw9ypzAO+sf0
/RR3w6RbKFfCs/mC/bdFWJsCAwEAAaOCAVowggFWMBIGA1UdEwEB/wQIMAYBAf8C
AQAwDgYDVR0PAQH/BAQDAgGGMDQGCCsGAQUFBwEBBCgwJjAkBggrBgEFBQcwAYYY
aHR0cDovL29jc3AuZGlnaWNlcnQuY29tMHsGA1UdHwR0MHIwN6A1oDOGMWh0dHA6
Ly9jcmwzLmRpZ2ljZXJ0LmNvbS9EaWdpQ2VydEdsb2JhbFJvb3RDQS5jcmwwN6A1
oDOGMWh0dHA6Ly9jcmw0LmRpZ2ljZXJ0LmNvbS9EaWdpQ2VydEdsb2JhbFJvb3RD
QS5jcmwwPQYDVR0gBDYwNDAyBgRVHSAAMCowKAYIKwYBBQUHAgEWHGh0dHBzOi8v
d3d3LmRpZ2ljZXJ0LmNvbS9DUFMwHQYDVR0OBBYEFA+AYRyCMWHVLyjnjUY4tCzh
xtniMB8GA1UdIwQYMBaAFAPeUDVW0Uy7ZvCj4hsbw5eyPdFVMA0GCSqGSIb3DQEB
CwUAA4IBAQAjPt9L0jFCpbZ+QlwaRMxp0Wi0XUvgBCFsS+JtzLHgl4+mUwnNqipl
5TlPHoOlblyYoiQm5vuh7ZPHLgLGTUq/sELfeNqzqPlt/yGFUzZgTHbO7Djc1lGA
8MXW5dRNJ2Srm8c+cftIl7gzbckTB+6WohsYFfZcTEDts8Ls/3HB40f/1LkAtDdC
2iDJ6m6K7hQGrn2iWZiIqBtvLfTyyRRfJs8sjX7tN8Cp1Tm5gr8ZDOo0rwAhaPit
c+LJMto4JQtV05od8GiG7S5BNO98pVAdvzr508EIDObtHopYJeS4d60tbvVS3bR0
j6tJLp07kzQoH3jOlOrHvdPJbRzeXDLz
-----END CERTIFICATE-----
```

プライバシー強化メール（PEM）インポートが正常に行われたことを確認します。証明書をインポートすると、確認メッセージが表示されます。

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **parameter-map type umbrella global**
4. **dnscrypt**
5. **token** *value*
6. **end**

DETAILED STEPS

	Command or Action	Purpose
ステップ 1	**enable** **Example:** `Device> enable`	特権 EXEC モードを有効にします。プロンプトが表示されたらパスワードを入力します。

	Command or Action	Purpose
ステップ 2	**configure terminal** **Example:** `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	**parameter-map type umbrella global** **Example:** `Device(config)# parameter-map type umbrella global`	パラメータマップタイプを umbrella モードに設定し、パラメータマップタイプ検査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	**dnscrypt** **Example:** `Device(config-profile)# dnscrypt`	デバイスで DNS パケット暗号化を有効にします。
ステップ 5	**token** *value* **Example:** `Device(config-profile)# token AABBA59A0BDE1485C912AFE472952641001EEECC`	Cisco Umbrella 登録サーバによって発行された API トークンを指定します。
ステップ 6	**end** **Example:** `Device(config-profile)# end`	パラメータ マップ タイプ検査コンフィギュレーション モードを終了して、特権 EXEC モードに戻ります。

Cisco Umbrella タグの登録

Before you begin

- Umbrella Connector を設定します。
- **umbrella in** コマンドを設定する前に **umbrella out** コマンドを設定します。登録は、ポート 443 がオープン状態にあり、既存のファイアウォールへのトラフィックのパススルーが許可される場合にのみ成功します。
- タグを指定して **umbrella in** コマンドを設定すると、デバイスは api.opendns.com を解決して登録プロセスを開始します。**ip name-server** コマンドを使用してネームサーバを設定し、デバイスで設定された **ip domain-lookup** コマンドを使用してドメインルックアップを設定して、FQDN を正常に解決します。

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **interface** *interface-type interface-number*
4. **umbrella out**
5. **exit**
6. **interface** *interface-type interface-number*
7. **umbrella in** *tag-name*
8. **end**

DETAILED STEPS

	Command or Action	Purpose
ステップ 1	**enable** **Example:** `Device> enable`	特権 EXEC モードを有効にします。プロンプトが表示されたらパスワードを入力します。
ステップ 2	**configure terminal** **Example:** `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** *interface-type interface-number* **Example:** `Device(config)# interface gigabitEthernet 1/1`	WAN インターフェイスを指定して、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	**umbrella out** **Example:** `Device(config-if)# umbrella out`	Umbrella クラウドサーバに接続するためのインターフェイスで Umbrella Connector を設定します。
ステップ 5	**exit** **Example:** `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を開始します。
ステップ 6	**interface** *interface-type interface-number* **Example:** `Device(config)# interface gigabitEthernet 1/2`	LAN インターフェイスを指定して、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	**umbrella in** *tag-name* **Example:**	クライアントに接続されているインターフェイスで Umbrella Connector を設定します。

	Command or Action	Purpose
	`Device(config-if)# umbrella in mydevice_tag`	• Umbrella タグの長さは 49 文字までです。 • タグを使用して **umbrella in** コマンドを設定すると、デバイスは Cisco Umbrella 統合サーバにタグを登録します。
ステップ 8	**end** **Example:** `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。

Cisco デバイスをパススルーサーバとして設定

ドメイン名を使用して、バイパスされるトラフィックを特定することができます。Cisco デバイスでは、正規表現形式でこれらのドメインを定義できます。デバイスによって横取りされた DNS クエリが、設定済みの正規表現の 1 つにマッチすると、このクエリは、Umbrella クラウドにリダイレクトされずに、指定された DNS サーバにバイパスされます。

SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **parameter-map type regex** *parameter-map-name*
4. **pattern** *expression*
5. **exit**
6. **parameter-map type umbrella global**
7. **token** *value*
8. **local-domain** *regex_param_map_name*
9. **end**

DETAILED STEPS

	Command or Action	Purpose
ステップ 1	**enable** **Example:** `Device> enable`	特権 EXEC モードを有効にします。プロンプトが表示されたらパスワードを入力します。
ステップ 2	**configure terminal** **Example:** `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。

	Command or Action	Purpose
ステップ 3	**parameter-map type regex** *parameter-map-name* **Example:** `Device(config)# parameter-map type regex dns_bypass`	パラメータマップタイプを指定されたトラフィックパターンに一致するように設定し、パラメータマップタイプ検査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	**pattern** *expression*
	Example:
	`Device(config-profile)# pattern www.cisco.com`
	`Device(config-profile)# pattern .*example.cisco.*`
		Umbrella クラウドをバイパスするために使用するローカルドメインまたは URL を設定します。

ステップ 5	**exit** **Example:** `Device(config-profile)# exit`	パラメータ マップ タイプ検査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 モードを開始します。
ステップ 6	**parameter-map type umbrella global** **Example:** `Device(config)# parameter-map type umbrella global`	パラメータマップタイプを umbrella モードに設定し、パラメータマップタイプ検査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	**token** *value* **Example:** `Device(config-profile)# token AADDD5FF6E510B28921A20C9B98EEEFF`	Cisco Umbrella 登録サーバによって発行された API トークンを指定します。
ステップ 8	**local-domain** *regex_param_map_name* **Example:** `Device(config-profile)# local-domain dns_bypass`	正規表現パラメータマップを Umbrella グローバル コンフィギュレーションにアタッチします。
ステップ 9	**end** **Example:** `Device(config-profile)# end`	パラメータ マップ タイプ検査コンフィギュレーション モードを終了して、特権 EXEC モードに戻ります。

Cisco Umbrella 統合の設定の確認

Cisco Umbrella 統合の設定を表示および確認するには、次のコマンドを任意の順序で使用します。

次に、**show umbrella config** コマンドの出力例を示します。

```
Device# show umbrella config
Umbrella Configuration
=======================
Token: EB74330C50767B6A63770EA6C3408DCF00282D8E
API-KEY: NONE
OrganizationID: 2633102
Local Domain Regex parameter-map name: NONE
DNSCrypt: Enabled
Public-key: B735:1140:206F:225D:3E2B:D822:D7FD:691E:A1C3:3CC8:D666:8D0C:BE04:BFAB:CA43:FB79
UDP Timeout: 5 seconds
Resolver address:
1. 208.67.220.220
2. 208.67.222.222
3. 2620:119:53::53
4. 2620:119:35::35
Umbrella Interface Config:
Number of interfaces with "umbrella out" config: 1
1. GigabitEthernet1/4
Mode : OUT
VRF : global(Id: 0)
Number of interfaces with "umbrella in" config: 2
1. GigabitEthernet1/9
Mode : IN
DCA : Disabled
Tag : IE_uniquetag
Device-id : 010a424c1597fe09
VRF : global(Id: 0)
2. GigabitEthernet2/3
Mode : IN
DCA : Disabled
Tag : IE_tag_2
Device-id : 010adaf012a36ad6
VRF : global(Id: 0)
Configured Umbrella Parameter-maps:
1. global
```

次に、**show umbrella deviceid** コマンドの出力例を示します。

```
Device# show umbrella deviceid

Device registration details
Interface Name Tag Status Device-id
GigabitEthernet1/9 IE_uniquetag 200 SUCCESS 010a424c1597fe09
GigabitEthernet2/3 IE_tag_2 200 SUCCESS 010adaf012a36ad6
```

次に、**show umbrella dnscrypt** コマンドの出力例を示します。

```
Device#show umbrella dnscrypt
DNSCrypt: Enabled
Public-key: B735:1140:206F:225D:3E2B:D822:D7FD:691E:A1C3:3CC8:D666:8D0C:BE04:BFAB:CA43:FB79
Certificate Update Status:
Last Successfull Attempt: 20:01:18 IST Dec 17 2019
```

```
Certificate Details:
Certificate Magic : DNSC
Major Version : 0x0001
Minor Version : 0x0000
Query Magic : 0x7163373861576F6F
Serial Number : 1574811744
Start Time : 1574811744 (05:12:24 IST Nov 27 2019)
End Time : 1606347744 (05:12:24 IST Nov 26 2020)
Server Public Key :
88B4:E44B:35E9:64B4:90BD:DABA:E825:A24B:0415:A08B:E19D:7DDB:87A3:3CD7:7EDF:8E2F
Client Secret Key Hash:
0FB9:520E:5228:FB2C:D521:1E9E:2ACB:AC3D:B520:A795:F54C:C608:604B:A410:17F1:1284
Client Public key :
E42F:507E:F052:72DD:1BC8:4857:2AE0:2F9F:ED87:1687:AAE4:095D:D933:48F0:5D60:3662
NM key Hash :
EDC3:25DD:4D21:103E:7E49:1EFA:75ED:4D6F:A450:107D:C6E8:1C41:9CF7:4039:FA89:2CED
```

次に、**show umbrella deviceid detailed** コマンドの出力例を示します。

```
Device# show umbrella deviceid detailed

Device registration details
1.GigabitEthernet1/9
Tag : IE_uniquetag
Device-id : 010a424c1597fe09
Description : Device Id recieved successfully
WAN interface : GigabitEthernet1/4
WAN VRF used : global(Id: 0)
2.GigabitEthernet2/3
Tag : IE_tag_2
Device-id : 010adaf012a36ad6
Description : Device Id recieved successfully
WAN interface : GigabitEthernet1/4
WAN VRF used : global(Id: 0)
```

次に、**show platform software dns-umbrella statistics** コマンドの出力例を示します。コマンド出力には、送信されたクエリの数、受信した応答の数などのトラフィック関連の情報が表示されます。

```
Device# show platform software dns-umbrella statistics

=======================================
Umbrella Statistics
=======================================
Total Packets : 7848
DNSCrypt queries : 3940
DNSCrypt responses : 0
DNS queries : 0
DNS bypassed queries(Regex) : 0
DNS responses(Umbrella) : 0
DNS responses(Other) : 3906
Aged queries : 34
Dropped pkts : 0
```

Cisco Umbrella 統合のトラブルシューティング

次のコマンドを使用して、Cisco Umbrella 統合機能の設定に関連する問題をトラブルシューティングできます。

表 *5 : Cisco Umbrella* 統合機能のデバッグコマンド

コマンド	目的
debug umbrella config	Umbrella 設定のデバッグを有効にします。
debug umbrella device-registration	Umbrella デバイス登録のデバッグを有効にします。
debug umbrella dnscrypt	Umbrella DNSCrypt 暗号化のデバッグを有効にします。

Windows マシンのコマンドプロンプト、または Linux マシンのターミナルウィンドウもしくはシェルから、**nslookup -type=txt debug.opendns.com** コマンドを実行します。**nslookup -type=txt debug.opendns.com** コマンドで指定する IP アドレスは、DNS サーバの IP アドレスである必要があります。

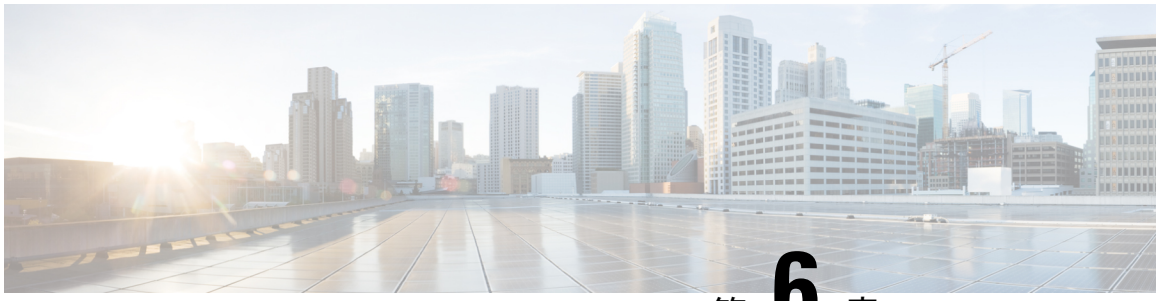
```
nslookup -type=txt debug.opendns.com 10.0.0.1
Server: 10.0.0.1
Address: 10.0.0.1#53
Non-authoritative answer:
debug.opendns.com text = "server r6.xx"
debug.opendns.com text = "device 010A826AAABB6C3D"
debug.opendns.com text = "organization id 1892929"
debug.opendns.com text = "remoteip 10.0.1.1"
debug.opendns.com text = "flags 436 0 6040 39FF000000000000000"
debug.opendns.com text = "originid 119211936"
debug.opendns.com text = "orgid 1892929"
debug.opendns.com text = "orgflags 3"
debug.opendns.com text = "actype 0"
debug.opendns.com text = "bundle 365396"
debug.opendns.com text = "source 10.1.1.1:36914"
debug.opendns.com text = "dnscrypt enabled (713156774457306E)"
```

Cisco Umbrella 統合の機能情報

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。

表 *6 : Cisco Umbrella* 統合の機能情報

機能名	リリース	機能情報
Cisco Umbrella 統合	Cisco IOS XE Amsterdam 17.2.1	Cisco Umbrella 統合機能により、Cisco デバイスを介して任意の DNS サーバに送信される DNS クエリを検査する、クラウドベースのセキュリティサービスを利用できるようになります。セキュリティ管理者は、FQDN へのトラフィックを許可または拒否するポリシーを Cisco Umbrella クラウドに設定します。



第 6 章

Web ベース認証の設定

Web ベース認証について

Web ベース認証の概要

IEEE 802.1x サプリカントが実行されていないホストシステムでエンドユーザを認証するには、Web 認証プロキシとして知られている Web ベース認証機能を使用します。

HTTP セッションを開始すると、Web ベース認証は、ホストからの受信 HTTP パケットを横取りし、ユーザに HTML ログインページを送信します。ユーザはクレデンシャルを入力します。このクレデンシャルは、Web ベース認証機能により、認証のために認証、許可、アカウンティング（AAA）サーバに送信されます。

認証が成功すると、Web ベース認証はログイン成功 HTML ページをホストに送信し、AAA サーバから返されたアクセス ポリシーを適用します。

認証に失敗した場合、Web ベース認証は、ログインの失敗を示す HTML ページをユーザに転送し、ログインを再試行するように、ユーザにプロンプトを表示します。最大試行回数を超過した場合、Web ベース認証は、ログインの期限切れを示す HTML ページをホストに転送し、このユーザは待機期間中、ウォッチ リストに載せられます。

（注） 中央 Web 認証リダイレクト用の HTTPS トラフィック インターセプションはサポートされていません。

（注） グローバル パラメータ マップ（method-type、custom、redirect）は、すべてのクライアントおよび SSID で同じ Web 認証方式（consent、web consent、webauth など）を使用するときにのみ使用する必要があります。これにより、すべてのクライアントが同じ Web 認証方式になります。

要件により、1 つの SSID に consent、別の SSID に webauth を使用する場合、名前付きパラメータ マップを 2 つ使用する必要があります。1 番目のパラメータ マップには consent を設定し、2 番目のパラメータ マップには webauth を設定する必要があります。

（注） Webauth クライアントの認証試行時に受信する traceback には、パフォーマンスや行動への影響はありません。これは、ACL アプリケーションの EPM に FFM が返信したコンテキストがすでにキュー解除済み（タイマーの有効期限切れの可能性あり）で、セッションが「未承認」になった場合にまれに発生します。

Web ページがホストされている場所に基づいて、ローカル Web 認証は次のように分類できます。

- 内部：ローカル Web 認証時に、コントローラの内部デフォルト HTML ページ（ログイン、成功、失敗、および期限切れ）が使用されます。
- カスタマイズ：ローカル Web 認証時に、カスタマイズされた Web ページ（ログイン、成功、失敗、および期限切れ）がコントローラにダウンロードされ、使用されます。
- 外部：組み込みまたはカスタム Web ページを使用する代わりに、外部 Web サーバ上でカスタマイズされた Web ページがホストされます。

（注） このリリースでは、外部 Web 認証はサポートされていません。

さまざまな Web 認証ページに基づき、Web 認証のタイプは次のように分類できます。

- *Webauth*：これが基本的な Web 認証です。この場合、コントローラはユーザ名とパスワードの入力が必要なポリシーページを提示します。ネットワークにアクセスするには、ユーザは正しいクレデンシャルを入力する必要があります。
- Consent または web-passthrough：この場合、コントローラは [Accept] ボタンが表示されたポリシーページを提示します。ネットワークにアクセスするには、ユーザは [Accept] ボタンをクリックする必要があります。
- *Webconsent*：これは webauth と consent の Web 認証タイプの組み合わせです。この場合、コントローラは [Accept] ボタンとともにユーザ名とパスワードが表示されたポリシーページを提示します。ネットワークにアクセスするには、ユーザは正しいクレデンシャルを入力して [Accept] ボタンをクリックする必要があります。

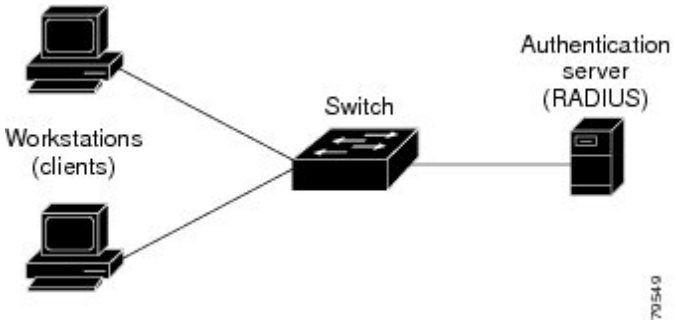
デバイスのロール

Web ベース認証では、ネットワーク上のデバイスに次のような固有の役割があります。

- クライアント：LAN およびサービスへのアクセスを要求し、スイッチからの要求に応答するデバイス（ワークステーション）。このワークステーションでは、Java Script が有効な HTML ブラウザが実行されている必要があります。
- 認証サーバ：クライアントを認証します。認証サーバはクライアントの識別情報を確認し、そのクライアントが LAN およびスイッチのサービスへのアクセスを許可されたか、あるいはクライアントが拒否されたのかをスイッチに通知します。
- スイッチ：クライアントの認証ステータスに基づいて、ネットワークへの物理アクセスを制御します。スイッチはクライアントと認証サーバとの仲介デバイス（プロキシ）として動作し、クライアントに識別情報を要求し、その情報を認証サーバで確認し、クライアントに応答をリレーします。

図 *6 : Web* ベース認証デバイスの役割

次の図は、ネットワーク上でのこれらのデバイスの役割を示します。



ホストの検出

スイッチは、検出されたホストに関する情報を格納するために、IP デバイス追跡 テーブルを維持します。

レイヤ 2 インターフェイスでは、Web ベース認証は、これらのメカニズムを使用して、IP ホストを検出します。

- ARP ベースのトリガー：ARP リダイレクト ACL により、Web ベース認証は、スタティック IP アドレス、またはダイナミック IP アドレスを持つホストを検出できます。
- ダイナミック ARP検査
- DHCP スヌーピング：スイッチがホストの DHCP バインディング エントリを作成するときに Web ベース認証が通知されます。

セッションの作成

Web ベース認証により、新しいホストが検出されると、次のようにセッションが作成されます。

- 例外リストをレビューします。

  ホスト IP が例外リストに含まれている場合、この例外リスト エントリからポリシーが適用され、セッションが確立されます。

- 認証バイパスをレビューします。

  ホストIPが例外リストに含まれていない場合、Webベース認証は応答しないホスト（NRH）要求をサーバに送信します。

  サーバの応答が access accepted であった場合、認証はこのホストにバイパスされます。セッションが確立されます。

- HTTP インターセプト ACL を設定します。

  NRH 要求に対するサーバの応答が access rejected であった場合、HTTP インターセプト ACLがアクティブ化され、セッションはホストからのHTTPトラフィックを待機します。

認証プロセス

Web ベース認証を有効にすると、次のイベントが発生します。

- ユーザが HTTP セッションを開始します。
- HTTP トラフィックが横取りされ、認証が開始されます。スイッチは、ユーザにログインページを送信します。ユーザはユーザ名とパスワードを入力します。スイッチはこのエントリを認証サーバに送信します。
- 認証に成功した場合、スイッチは認証サーバからこのユーザのアクセスポリシーをダウンロードし、アクティブ化します。ログインの成功ページがユーザに送信されます
- 認証に失敗した場合は、スイッチはログインの失敗ページを送信します。ユーザはログインを再試行します。失敗の回数が試行回数の最大値に達した場合、スイッチはログイン期限切れページを送信します。このホストはウォッチ リストに入れられます。ウォッチ リストのタイム アウト後、ユーザは認証プロセスを再試行することができます。
- 認証サーバがスイッチに応答せず、AAA 失敗ポリシーが設定されている場合、スイッチはホストに失敗アクセスポリシーを適用します。ログインの成功ページがユーザに送信されます
- ホストがレイヤ 2 インターフェイス上の ARP プローブに応答しなかった場合、またはホストがレイヤ 3 インターフェイスでアイドル タイムアウト内にトラフィックを送信しなかった場合、スイッチはクライアントを再認証します。
- ホストがレイヤ 2 インターフェイス上の ARP プローブに応答しない場合、スイッチはクライアントを再認証します。
- この機能は、ダウンロードされたタイムアウト、またはローカルに設定されたセッションタイムアウトを適用します。
- Termination-Action が RADIUS である場合、この機能は、サーバに NRH 要求を送信します。Termination-Action は、サーバからの応答に含まれます。

- Termination-Action がデフォルトである場合、セッションは廃棄され、適用されたポリシーは削除されます。

ローカル Web 認証バナー

Web 認証を使用して、デフォルトのカスタマイズ済み Web ブラウザ バナーを作成して、スイッチにログインしたときに表示するようにできます。

このバナーは、ログインページと認証結果ポップアップページの両方に表示されます。デフォルトのバナー メッセージは次のとおりです。

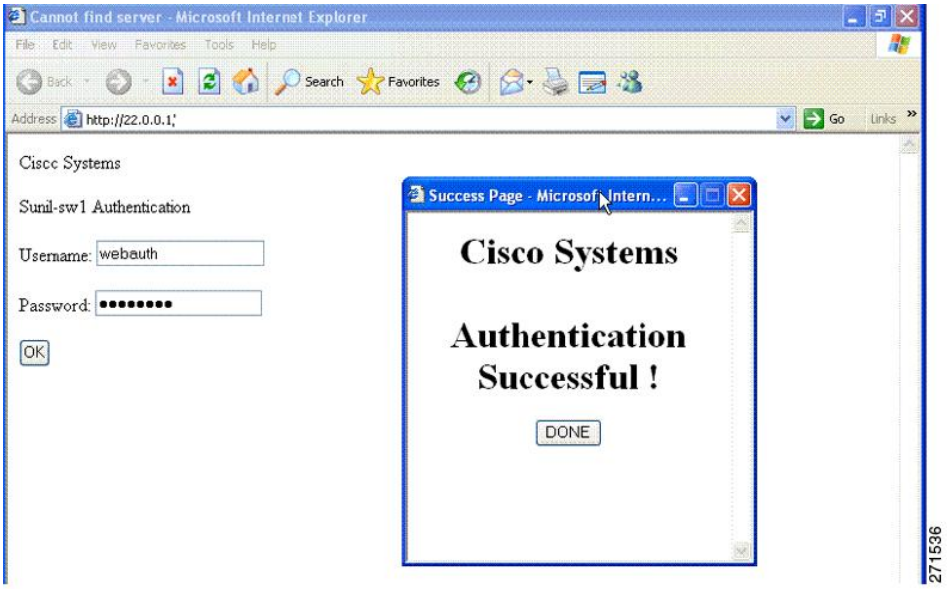
- 認証成功
- 認証失敗
- 認証期限切れ

ローカル Web 認証バナーは、次のように設定できます。

- レガシー モード：**ip admission auth-proxy-banner http** グローバル コンフィギュレーション コマンドを使用します。
- 新スタイル モード：**parameter-map type webauth global banner** グローバル コンフィギュレーション コマンドを使用します。

ログイン ページには、デフォルトのバナー、*Cisco Systems*、および *Switch host-name Authentication* が表示されます。*Cisco Systems* は認証結果ポップアップ ページに表示されます。

図 7: 認証成功バナー



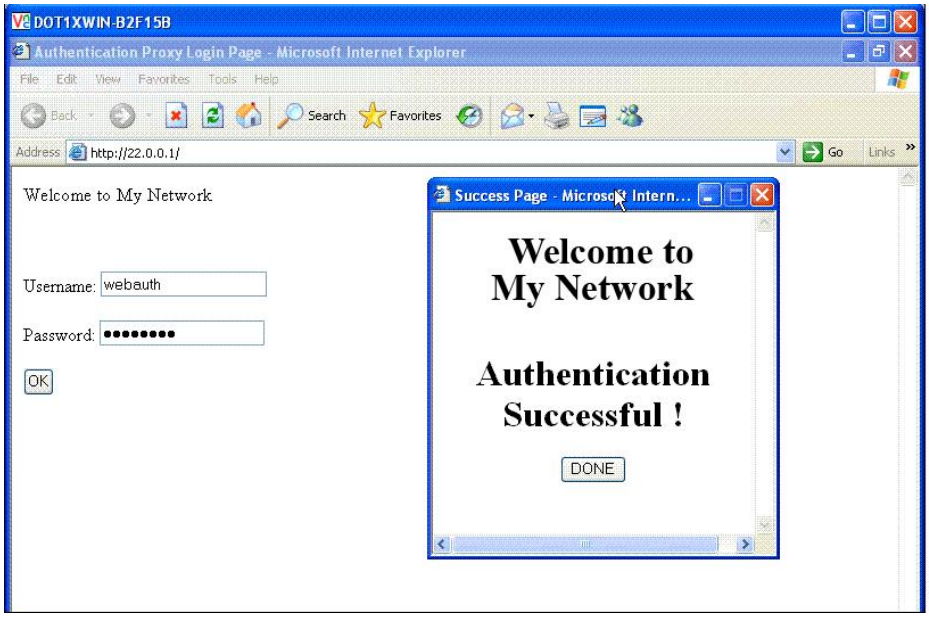
バナーは次のようにカスタマイズ可能です。

- スイッチ名、ルータ名、または会社名などのメッセージをバナーに追加する。

- レガシーモード： **ip admission auth-proxy-banner http** *banner-text* グローバル コンフィギュレーション コマンドを使用します。
- 新スタイル モード：**parameter-map type webauth global banner** グローバル コンフィギュレーション コマンドを使用します。

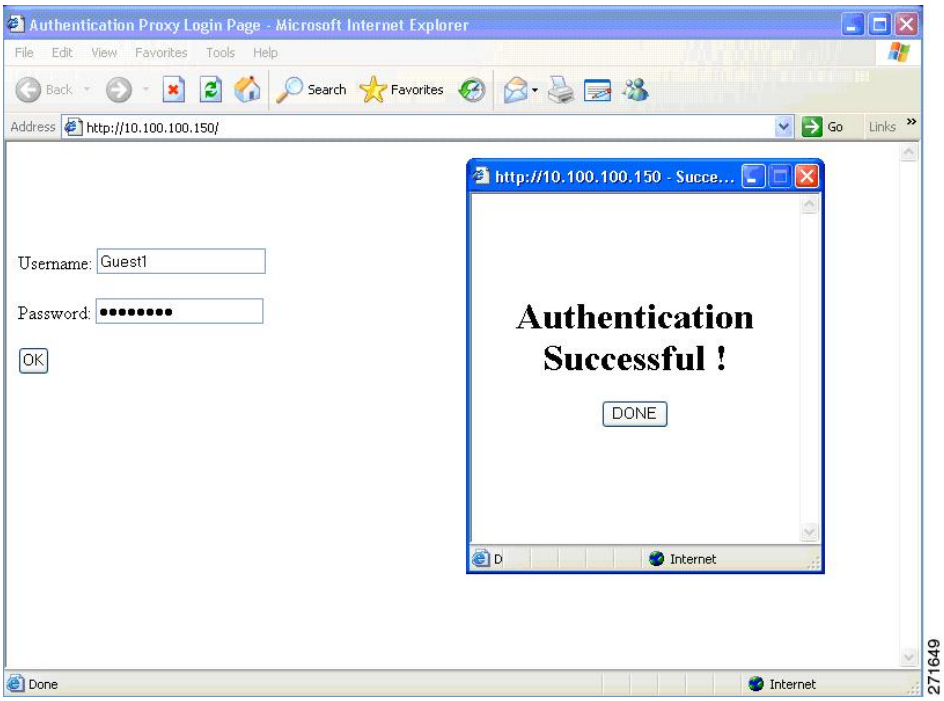
- ロゴまたはテキスト ファイルをバナーに追加する。
  - レガシーモード：**ip admission auth-proxy-banner http** *file-path* グローバル コンフィギュレーション コマンドを使用します。
  - 新スタイル モード：**parameter-map type webauth global banner** グローバル コンフィギュレーション コマンドを使用します。

図 *8* : カスタマイズされた *Web* バナー



バナーが有効にされていない場合、Web認証ログイン画面にはユーザ名とパスワードのダイアログボックスだけが表示され、スイッチにログインしたときにはバナーは表示されません。

図 *9* : バナーが表示されていないログイン画面



Web 認証カスタマイズ可能な Web ページ

Web ベース認証プロセスでは、スイッチ内部の HTTP サーバは、認証中のクライアントに配信される 4 種類の HTML ページをホストします。サーバはこれらのページを使用して、ユーザに次の 4 種類の認証プロセス ステートを通知します。

- ログイン：資格情報が要求されています。
- 成功：ログインに成功しました。
- 失敗：ログインに失敗しました。
- 期限切れ：ログインの失敗回数が多すぎて、ログイン セッションが期限切れになりました。

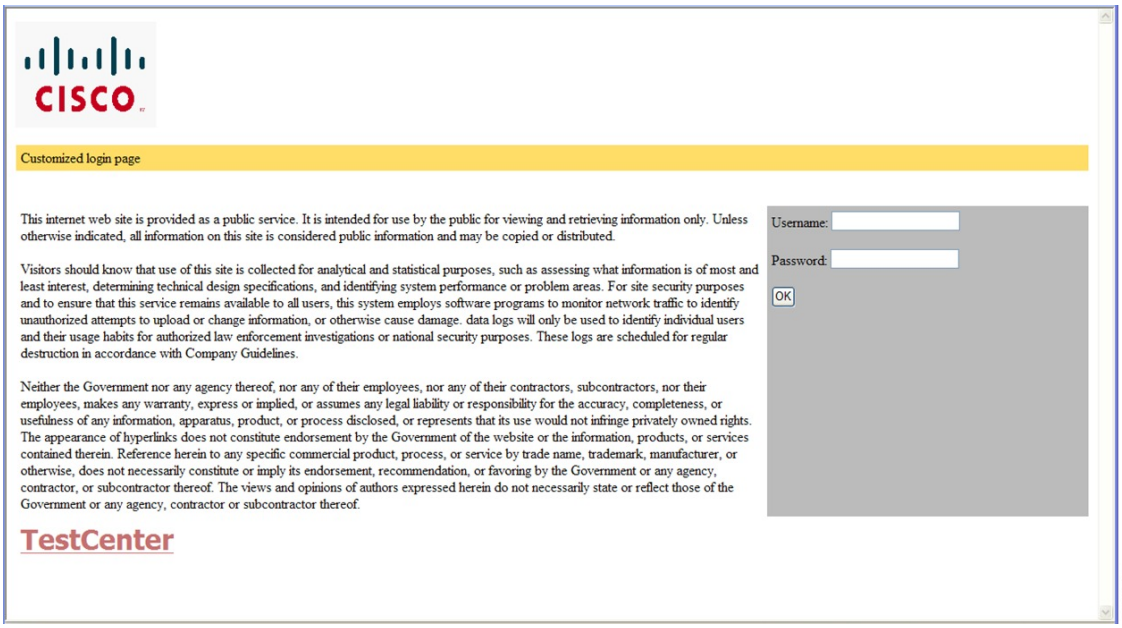
ガイドライン

- デフォルトの内部 HTML ページの代わりに、独自の HTML ページを使用することができます。
- ロゴを使用することもできますし、ログイン、成功、失敗、および期限切れ Web ページでテキストを指定することもできます。
- バナー ページで、ログイン ページのテキストを指定できます。
- これらのページは、HTML で記述されています。

- 成功ページには、特定の URL にアクセスするための HTML リダイレクト コマンドを記入する必要があります。
- この URL 文字列は有効な URL（例：http://www.cisco.com）でなければなりません。不完全な URL は、Web ブラウザで、「ページが見つかりません」またはこれに類似するエラーの原因となる可能性があります。
- HTTP 認証で使用される Web ページを設定する場合、これらのページには適切な HTML コマンド（例：ページのタイムアウトを設定、暗号化されたパスワードの設定、同じページが 2 回送信されていないことの確認など）を記入する必要があります.
- 設定されたログイン フォームが有効な場合、特定の URL にユーザをリダイレクトする CLI コマンドは使用できません。管理者は、Web ページにリダイレクトが設定されていることを保証する必要があります。
- 認証後、特定の URL にユーザをリダイレクトする CLI コマンドを入力してから、Web ページを設定するコマンドを入力した場合、特定の URL にユーザをリダイレクトする CLI コマンドは効力を持ちません。
- 設定された Web ページは、スイッチのブート フラッシュ、またはフラッシュにコピーできます。
- ログイン ページを 1 つのフラッシュ上に、成功ページと失敗ページを別のフラッシュ（たとえば、スタック マスター、またはメンバのフラッシュ）にすることができます。
- 4 ページすべてを設定する必要があります。
- Web ページを使ってバナー ページを設定した場合、このバナー ページには効果はありません。
- システム ディレクトリ（たとえば、flash、disk0、disk）に保存されていて、ログイン ページに表示する必要のあるロゴ ファイル（イメージ、フラッシュ、オーディオ、ビデオなど）すべてには、必ず、*web_auth_<filename>* の形式で名前をつけてください。
- 設定された認証プロキシ機能は、HTTP と SSL の両方をサポートしています。

デフォルトの内部 HTML ページの代わりに、自分の HTML ページを使用することができます。認証後のユーザのリダイレクト先で、内部成功ページの代わりとなる URL を指定することもできます。

図 *10* : カスタマイズ可能な認証ページ



認証プロキシ Web ページの注意事項

カスタマイズされた認証プロキシ Web ページを設定する際には、次の注意事項に従ってください。

- カスタム Web ページ機能を有効にするには、カスタム HTML ファイルを 4 個すべて指定します。指定したファイルの数が 4 個未満の場合、内部デフォルト HTML ページが使用されます。
- これら 4 個のカスタム HTML ファイルは、スイッチのフラッシュ メモリ内に存在しなければなりません。各 HTML ファイルの最大サイズは 8 KB です。
- カスタム ページ上のイメージはすべて、アクセス可能は HTTP サーバ上に存在しなければなりません。インターセプト ACL は、管理ルール内で設定します。
- カスタム ページからの外部リンクはすべて、管理ルール内でのインターセプト ACL の設定を必要とします。
- 有効な DNS サーバにアクセスするには、外部リンクまたはイメージに必要な名前解決で、管理ルール内にインターセプト ACL を設定する必要があります。
- カスタム Web ページ機能が有効に設定されている場合、設定された auth-proxy-banner は使用されません。
- カスタム Web ページ機能が有効に設定されている場合、ログインの成功に対するリダイレクション URL は使用できません。
- カスタム ファイルの指定を解除するには、このコマンドの **no** 形式を使用します。

カスタム ログイン ページはパブリック Web フォームであるため、このページについては、次の注意事項に従ってください。

- ログイン フォームは、ユーザによるユーザ名とパスワードの入力を受け付け、これらを **uname** および **pwd** として示す必要があります。
- カスタム ログイン ページは、ページ タイムアウト、暗号化されたパスワード、冗長送信の防止など、Web フォームに対するベスト プラクティスに従う必要があります。

その他の機能と Web ベース認証の相互作用

ポート セキュリティ

Webベース認証とポートセキュリティは、同じポートに設定できます。Webベース認証はポートを認証し、ポート セキュリティは、クライアントの MAC アドレスを含むすべての MAC アドレスに対するネットワーク アクセスを管理します。この場合、このポートを介してネットワークへアクセスできるクライアントの数とグループを制限できます。

LAN ポート IP

LANポートIP（LPIP）とレイヤ2Webベース認証は、同じポートに設定できます。ホストは、まずWebベース認証、次にLPIPポスチャ検証を使用して認証されます。LPIPホストポリシーは、Web ベース認証のホスト ポリシーに優先されます。

Web ベース認証のアイドル時間が満了すると、NAC ポリシーは削除されます。ホストが認証され、ポスチャが再度検証されます。

ゲートウェイ IP

VLANのいずれかのスイッチポートでWebベース認証が設定されている場合、レイヤ3VLANインターフェイス上にゲートウェイ IP（GWIP）を設定することはできません。

Web ベース認証はゲートウェイ IP と同じレイヤ 3 インターフェイスに設定できます。ソフトウェアで、両方の機能のホストポリシーが適用されます。GWIPホストポリシーは、Webベース認証のホスト ポリシーに優先されます。

ACL

インターフェイスで VLAN ACL、または Cisco IOS ACL を設定した場合、ACL は、Web ベース認証のホスト ポリシーが適用された後だけ、ホスト トラフィックに適用されます。

レイヤ2Webベース認証では、ポートに接続されたホストからの受信トラフィックについて、ポートACL（PACL）をデフォルトのアクセスポリシーとして設定することが必須ではないものの、より安全です。認証後、Webベース認証のホストポリシーは、PACLに優先されます。ポートに設定された ACL がなくても、ポリシー ACL はセッションに適用されます。

MAC ACL と Web ベース認証を同じインターフェイスに設定することはできません。

アクセス VLAN が VACL キャプチャ用に設定されているポートには Web ベース認証は設定できません。

コンテキストベース アクセス コントロール

コンテキストベース アクセス コントロール（CBAC）が、ポート VLAN のレイヤ 3 VLAN インターフェイスで設定されている場合、レイヤ2ポートでWebベース認証は設定できません。

EtherChannel

Web ベース認証は、レイヤ 2 EtherChannel インターフェイス上に設定できます。Web ベース認証設定は、すべてのメンバ チャネルに適用されます。

Web ベース認証の設定方法

デフォルトの Web ベース認証の設定

次の表に、デフォルトの Web ベース認証の設定を示しています。

表 *7:* デフォルトの *Web* ベース認証の設定

機能	デフォルト設定
AAA	無効
RADIUS サーバ • IP アドレス • UDP 認証ポート • キー	• 指定なし • 指定なし
無活動タイムアウトのデフォルト値	3600 秒
無活動タイムアウト	有効

Web ベース認証の設定に関する注意事項と制約事項

- Web ベース認証は受信時だけの機能です。
- Web ベース認証は、アクセス ポートだけで設定できます。Web ベース認証は、トランク ポート、EtherChannel メンバ ポート、またはダイナミック トランク ポートではサポートされていません。
- スイッチが特定のホストまたは Web サーバにクライアントをリダイレクトしてログイン メッセージを表示する場合、外部 Web 認証はサポートされません。

- スタティックな ARP キャッシュが割り当てられているレイヤ 2 インターフェイス上のホストは認証できません。これらのホストは ARP メッセージを送信しないため、Web ベース認証機能では検出されません。
- デフォルトでは、スイッチの IP デバイス追跡機能は無効にされています。Web ベース認証を使用するには、IP デバイス追跡機能を有効にする必要があります。
- Web ベース認証を使用するには、SISF ベースのデバイス追跡を有効にする必要があります。デフォルトでは、SISF ベースのデバイス追跡はスイッチで無効になっています。
- スイッチ HTTP サーバを実行するには、IP アドレスを少なくとも 1 つ設定する必要があります。また、各ホスト IP アドレスに到達するようにルートを設定する必要もあります。HTTP サーバは、ホストに HTTP ログイン ページを送信します。
- 2 ホップ以上離れたところにあるホストでは、STP トポロジの変更により、ホスト トラフィックの到着するポートが変わってしまった場合、トラフィックが停止する可能性があります。これは、レイヤ 2（STP）トポロジの変更後に、ARP および DHCP の更新が送信されていない場合に発生します。
- Web ベース認証は、ダウンロード可能なホスト ポリシーとして、VLAN 割り当てをサポートしていません。
- IPv6 Web ベース認証はサポートされていません。
- Web ベース認証および Network Edge Access Topology（NEAT）は、相互に排他的です。インターフェイス上で NEAT が有効の場合、Web ベース認証を使用できず、インターフェイス上で Web ベース認証が実行されている場合は、NEAT を使用できません。
- 仮想 IP 設定はサポートされていません。この制限により、ログアウトページはサポートされていません。
- スイッチから RADIUS サーバへの通信の設定に使用される次の RADIUS セキュリティ サーバ設定を確認します。
  - ホスト名
  - ホスト IP アドレス
  - ホスト名と特定の UDP ポート番号
  - IP アドレスと特定の UDP ポート番号

  IP アドレスと UDP ポート番号の組み合わせによって、一意の ID が作成され、サーバの同一 IP アドレス上にある複数の UDP ポートに RADIUS 要求を送信できるようになります。同じ RADIUS サーバ上の異なる 2 つのホスト エントリに同じサービス（たとえば認証）を設定した場合、2 番めに設定されたホスト エントリは、最初に設定されたホスト エントリのフェールオーバー バックアップとして動作します。RADIUS ホスト エントリは、設定した順序に従って選択されます。
- RADIUS サーバ パラメータを設定する場合は、次の点に注意してください。
  - 別のコマンドラインに、**key** *string* を指定します。

- **key** *string* には、スイッチと、RADIUS サーバ上で動作する RADIUS デーモンとの間で使用する、認証および暗号キーを指定します。キーは、RADIUS サーバで使用する暗号化キーに一致するテキスト ストリングでなければなりません。
- **key** *string* を指定する場合、キーの中間、および末尾にスペースを使用します。キーにスペースを使用する場合は、引用符がキーの一部分である場合を除き、引用符でキーを囲まないでください。キーはRADIUSデーモンで使用する暗号に一致している必要があります。
- すべてのRADIUSサーバについて、タイムアウト、再送信回数、および暗号キー値をグローバルに設定するには、**radius-server host** グローバル コンフィギュレーション コマンドを使用します。これらのオプションをサーバ単位で設定するには、**radius-server timeout**、radius-server transmit、および radius-server key グローバル コンフィギュレーション コマンドを使用します。

（注） RADIUSサーバでは、スイッチのIPアドレス、サーバとスイッチで共有されるkey string、およびダウンロード可能なACL（DACL）などの設定を行う必要があります。詳細については、RADIUSサーバのマニュアルを参照してください。

認証ルールとインターフェイスの設定

認証ルールおよびインターフェイスを設定するには、次の手順を実行します。

始める前に

SISFベースのデバイス追跡は、web認証の前提条件です。デバイス追跡をプログラムまたは手動で有効にしていることを確認します。

詳細については、「*SISF* ベースの追跡の設定」を参照してください。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip admission name** *name* **proxy http**
4. **interface** *type slot/port*
5. **ip access-group** *name*
6. **ip admission name**
7. **exit**
8. **show ip admission**

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	**ip admission name** *name* **proxy http** 例： `Device(config)# ip admission name webauth1 proxy http`	Web ベース許可の認証ルールを設定します。
ステップ 4	**interface** *type slot/port* 例： `Device(config)# interface gigabitethernet 1/0/1`	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、Web ベース認証を有効にする受信レイヤ 2 またはレイヤ 3 インターフェイスを指定します。 *type* には、FastEthernet、GigabitEthernet、または TenGigabitEthernet を指定できます。
ステップ 5	**ip access-group** *name* 例： `Device(config-if)# ip access-group webauthag`	デフォルト ACL を適用します。
ステップ 6	**ip admission name** 例： `Device(config)# ip admission name`	インターフェイスの Web ベース認可の認証ルールを設定します。
ステップ 7	**exit** 例： `Device# exit`	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 モードに戻ります。
ステップ 8	**show ip admission** 例：	ネットワークアドミッションのキャッシュエントリと Web 認証セッションに関する情報を表示します。

	コマンドまたはアクション	目的
	`Device# show ip admission`	

AAA 認証の設定

VTY 回線で方式リストを設定する場合、対応する方式リストを AAA 設定に追加する必要があります。

```
Device(config)# line vty 0 4
Device(config-line)# authorization commands 15 list1
Device(config-line)# exit
Device(config)# aaa authorization commands 15 list1 group tacacs+
```

VTY 回線で方式リストを設定しない場合、デフォルトの方式リストを AAA 設定に追加する必要があります。

```
Device(config)# line vty 0 4
Device(config-line)# exit
Device(config)# aaa authorization commands 15 default group tacacs+
```

AAA 認証を設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **aaa new-model**
4. **aaa authentication login default group** {**tacacs+** | **radius**}
5. **aaa authorization auth-proxy default group** {**tacacs+** | **radius**}
6. **tacacs server** *server-name*
7. **address** {**ipv4 | ipv6**} *ip address*
8. **key** 文字列
9. **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：	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。

	コマンドまたはアクション	目的
	`Device# configure terminal`	
ステップ 3	**aaa new-model** 例： `Device(config)# aaa new-model`	AAA 機能を有効にします。
ステップ 4	**aaa authentication login default group** {**tacacs+** | **radius**} 例： `Device(config)# aaa authentication login default group tacacs+`	ログイン時の認証方法のリストを定義します。 **named_authentication_list** は、31 文字未満の名前を示します。 **AAA_group_name** はサーバグループ名を示します。サーバグループ **server_name** をその先頭で定義する必要があります。
ステップ 5	**aaa authorization auth-proxy default group** {**tacacs+** | **radius**} 例： `Device(config)# aaa authorization auth-proxy default group tacacs+`	Web ベース許可の許可方式リストを作成します。
ステップ 6	**tacacs server** *server-name* 例： `Device(config)# tacacs server yourserver`	AAA サーバを指定します。
ステップ 7	**address** {**ipv4 | ipv6**} *ip address* 例： `Device(config-server-tacacs)# address ipv4 10.0.1.12`	TACACS サーバの IP アドレスを設定します。
ステップ 8	**key** 文字列 例： `Device(config-server-tacacs)# key cisco123`	スイッチと TACACS サーバとの間で使用される許可および暗号キーを設定します。
ステップ 9	**end** 例：	TACACS サーバモードを終了し、特権 EXEC モードに戻ります。

	コマンドまたはアクション	目的
	`Device(config-server-tacacs)# end`	

スイッチ/RADIUS サーバ間通信の設定

RADIUS サーバのパラメータを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip radius source-interface vlan** *vlan interface number*
4. **radius server** *server name*
5. **address** {**ipv4 | ipv6**} *ip address*
6. **key** *string*
7. **exit**
8. **radius-server dead-criteria tries** *num-tries*
9. **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	**ip radius source-interface vlan** *vlan interface number* 例： `Device(config)# ip radius source-interface vlan 80`	RADIUS パケットが、指定されたインターフェイスの IP アドレスを含むように指定します。
ステップ 4	**radius server** *server name* 例： `Device(config)# radius server rsim address ipv4`	（任意）RADIUS サーバの IP アドレスを指定します。

	コマンドまたはアクション	目的
	`172.16.0.1`	
ステップ 5	**address** {**ipv4** | **ipv6**} *ip address* 例： `Device(config-radius-server)# address ipv4 10.0.1.2 auth-port 1550 acct-port 1560`	RADIUS サーバの IP アドレスを設定します。
ステップ 6	**key** *string* 例： `Device(config-radius-server)# key rad123`	（任意）RADIUS サーバ上で動作する RADIUS デーモンとスイッチの間で使用する認証および暗号キーを指定します。
ステップ 7	**exit** 例： `Device(config-radius-server)# exit`	RADIUS サーバモードを終了して、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 8	**radius-server dead-criteria tries** *num-tries* 例： `Device(config)# radius-server dead-criteria tries 30`	RADIUS サーバに送信されたメッセージへの応答がない場合に、このサーバが非アクティブであると見なすまでの送信回数を指定します。指定できる *num-tries* の範囲は 1 ～ 100 です。
ステップ 9	**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 モードに戻ります。

HTTP サーバの設定

Web ベース認証を使用するには、device で HTTP サーバを有効にする必要があります。このサーバは HTTP または HTTPS のいずれかについて有効にできます。

（注） Apple の疑似ブラウザは、**ip http secure-server** コマンドを設定するだけでは開きません。**ip http server** コマンドも設定する必要があります。

HTTP または HTTPS のいずれかについてサーバを有効にするには、次の手順に従います。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip http server**
4. **ip http secure-server**
5. **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	**ip http server** 例： `Device(config)# ip http server`	HTTP サーバを有効にします。Web ベース認証機能は、HTTP サーバを使用してホストと通信し、ユーザ認証を行います。
ステップ 4	**ip http secure-server** 例： `Device(config)# ip http secure-server`	HTTPS を有効にします。 カスタム認証プロキシ Web ページを設定するか、成功ログインのリダイレクション URL を指定します。 （注） **ip http secure-server** コマンドを入力したときに、セキュア認証が確実に行われるようにするには、ユーザが HTTP 要求を送信した場合でも、ログインページは必ず HTTPS（セキュア HTTP）形式になるようにします。
ステップ 5	**end** 例： `Device# 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 モードに戻ります。

認証プロキシ Web ページのカスタマイズ

Web ベースの認証中、のデフォルト HTML ページではなく、代わりの HTML ページがユーザに表示されるように、Web 認証を設定できます。

カスタム認証プロキシ Web ページの使用を指定するには、次の手順を実行してください。

始める前に

device のフラッシュ メモリにカスタム HTML ファイルを保存します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip admission proxy http login page file** *device:login-filename*
4. **ip admission proxy http success page file** *device:success-filename*
5. **ip admission proxy http failure page file** *device:fail-filename*
6. **ip admission proxy http login expired page file** *device:expired-filename*
7. **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	**ip admission proxy http login page file** *device:login-filename* 例： `Device(config)# ip admission proxy http login page file disk1:login.htm`	device のメモリ ファイル システム内で、デフォルトのログイン ページの代わりに使用するカスタム HTML ファイルの場所を指定します。*device:* はフラッシュ メモリです。
ステップ 4	**ip admission proxy http success page file** *device:success-filename* 例： `Device(config)# ip admission proxy http success`	デフォルトのログイン成功ページの代わりに使用するカスタム HTML ファイルの場所を指定します。

	コマンドまたはアクション	目的
	`page file disk1:success.htm`	
ステップ 5	**ip admission proxy http failure page file** *device:fail-filename* 例： `Device(config)# ip admission proxy http fail page file disk1:fail.htm`	デフォルトのログイン失敗ページの代わりに使用するカスタム HTML ファイルの場所を指定します。
ステップ 6	**ip admission proxy http login expired page file** *device:expired-filename* 例： `Device(config)# ip admission proxy http login expired page file disk1:expired.htm`	デフォルトのログイン失効ページの代わりに使用するカスタム HTML ファイルの場所を指定します。
ステップ 7	**end** 例： `Device# 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 モードに戻ります。

Web ベース認証パラメータの設定

クライアントが待機時間中にウォッチ リストに掲載されるまで許容される失敗ログイン試行の最大回数を設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip admission max-login-attempts** *number*
4. **exit**

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）。

	コマンドまたはアクション	目的
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 admission max-login-attempts** *number* 例： `Device(config)# ip admission max-login-attempts 10`	失敗ログイン試行の最大回数を設定します。指定できる範囲は 1 ～ 2147483647 回です。デフォルトは 5 分です。
ステップ 4	**exit** 例： `Device# exit`	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 モードに戻ります。

Web ベース認証ローカル バナーの設定

Web 認証が設定されているスイッチにローカル バナーを設定するには、次の手順を実行します。

手順の概要

1. **enable**
2. **configure terminal**
3. **ip admission auth-proxy-banner http** [*banner-text* | *file-path*]
4. **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。

	コマンドまたはアクション	目的
ステップ 3	**ip admission auth-proxy-banner http** [*banner-text* | *file-path*] 例： `Device(config)# ip admission auth-proxy-banner http C My Switch C`	ローカル バナーを有効にします。 （任意）*C banner-text C*（*C* は区切り文字）、またはバナーに表示されるファイル（たとえば、ロゴまたはテキスト ファイル）のファイル パスを入力して、カスタム バナーを作成します。
ステップ 4	**end** 例： `Device# 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 モードに戻ります。

Web ベース認証キャッシュ エントリの削除

Web ベース認証キャッシュ エントリを削除するには、次の手順を実行します。

手順の概要

1. **enable**
2. **clear ip auth-proxy cache** {* | *host ip address*}
3. **clear ip admission cache** {* | *host ip address*}

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	**clear ip auth-proxy cache** {*	*host ip address*} 例： `Device# clear ip auth-proxy cache 192.168.4.5`	Delete 認証プロキシ エントリを削除します。キャッシュ エントリすべてを削除するには、アスタリスクを使用します。シングル ホストのエントリを削除するには、具体的な IP アドレスを入力します。
ステップ 3	**clear ip admission cache** {*	*host ip address*} 例： `# clear ip admission cache 192.168.4.5`	Delete 認証プロキシ エントリを削除します。キャッシュ エントリすべてを削除するには、アスタリスクを使用します。シングル ホストのエントリを削除するには、具体的な IP アドレスを入力します。

Web ベース認証の確認

すべてのインターフェイスまたは特定のポートに対する Web ベース認証設定を表示するには、このトピックのコマンドを使用します。

表 *8*: 特権 *EXEC* 表示コマンド

コマンド	目的
show authentication sessions method webauth	FastEthernet、ギガビット イーサネット、または 10 ギガビット イーサネットのすべてのインターフェイスに対する Web ベースの認証設定を表示します。
show authentication sessions interface *type slot/port*[**details**]	FastEthernet、ギガビット イーサネット、または 10 ギガビット イーサネットの特定のインターフェイスに対する Web ベースの認証設定を表示します。 セッション認識型ネットワーク モードでは、**show access-session interface** コマンドを使用します。

Web ベース認証に関するその他の参考資料

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』

シスコのテクニカル サポート

説明	リンク
シスコのサポート Web サイトでは、シスコの製品やテクノロジーに関するトラブルシューティングにお役立ていただけるように、マニュアルやツールをはじめとする豊富なオンライン リソースを提供しています。 お使いの製品のセキュリティ情報や技術情報を入手するために、Cisco Notification Service（Field Notice からアクセス）、Cisco Technical Services Newsletter、Really Simple Syndication（RSS）フィードなどの各種サービスに加入できます。 シスコのサポート Web サイトのツールにアクセスする際は、Cisco.com のユーザ ID およびパスワードが必要です。	http://www.cisco.com/support