CHAPTER 5

トラフィックの宛先変更の設定

この章では、Cisco Anomaly Guard Module（Guard モジュール）の宛先変更プロセスについて説明します。この章は、次の項で構成されています。

概要

Cisco Anomaly Guard Module（Guard モジュール）は、分散型アップストリーム構成に ISP/MSP/ バックボーン レベルで配置して、ネットワーク全体を保護するための高パフォーマンス ネットワーク サービス モジュールです。

Guard モジュールは Cisco IOS アプリケーション モジュールの 1 つで、次の製品のどちらかに設置できます。

- Supervisor Engine 720（SUP720）、または Multilayer Switch Feature Card 2（MSFC2; マルチレイヤ スイッチ フィーチャ カード 2）を備えた Supervisor Engine 2（SUP2）が搭載された、Cisco Catalyst 6500 シリーズ スイッチ。Catalyst 6500 で Guard モジュールをサポートするには、IOS 12.2(18)SXD3 以降が必要です。
- SUP720 が搭載された Cisco 7600 シリーズ ルータ。7600 シリーズ ルータで Guard モジュールをサポートするには、IOS 12.2(18)SXE 以降が必要です。

攻撃が検出されると、攻撃対象ゾーンのトラフィックのみが宛先変更され、Guard モジュールに送られます。Guard モジュールは、データ フローを分析します。DDoS 要素はすべてブロックされ、宛先変更されたストリームから悪意のあるパケットが削除されます。一方、正当なトラフィックは元の宛先に転送され、継続的に目的のゾーンに流されます。

ラーニング プロセスをアクティブにすると、スーパーバイザ エンジンがゾーン トラフィックを Guard モジュールに宛先変更します。Guard モジュールは、トラフィックを分析して、保護ポリシーを作成します。その後、トラフィックをゾーンのメイン トラフィック パスに戻します（変更は加えません）。

ゾーン トラフィックを Guard モジュールに宛先変更してからメインのデータ パスに戻すサイクル全体は、宛先変更プロセスと呼ばれます。

攻撃の疑いがない場合は、宛先変更プロセスをアクティブにする必要はなく、Guard モジュールはトラフィックを監視しません。

この項では、次のトピックについて取り上げます。

トラフィックの宛先変更

IP トラフィックの宛先変更では 2 つのタスクが実行されます。1 つまたはそれ以上のゾーンのトラフィックを Guard モジュールに宛先変更するタスクと、正当なトラフィックを Guard モジュールから元のデータ パスおよびゾーンに戻すタスクです。

宛先変更プロセスは、次の 2 つの処理で構成されています。

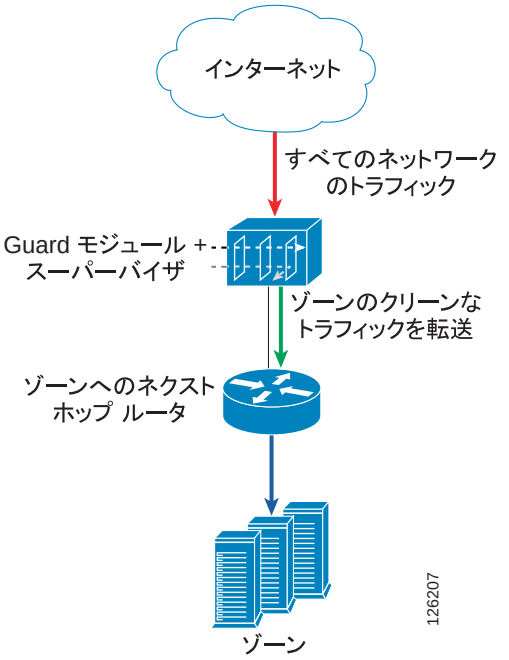
- **ハイジャック**：Guard　モジュールがゾーンを保護している場合、Guard　モジュールはゾーン トラフィックのルーティングを変更して、トラフィックが通常のスーパーバイザ エンジンのオンボード ルーティング テーブルをバイパスして Guard モジュールに流れるようにします。
- **注入**：Guard モジュールは、正当なトラフィックを元のデータ パスに戻します。

ネットワーク設定

Guard モジュールは、次のネットワーク設定のどちらかに設置できます。

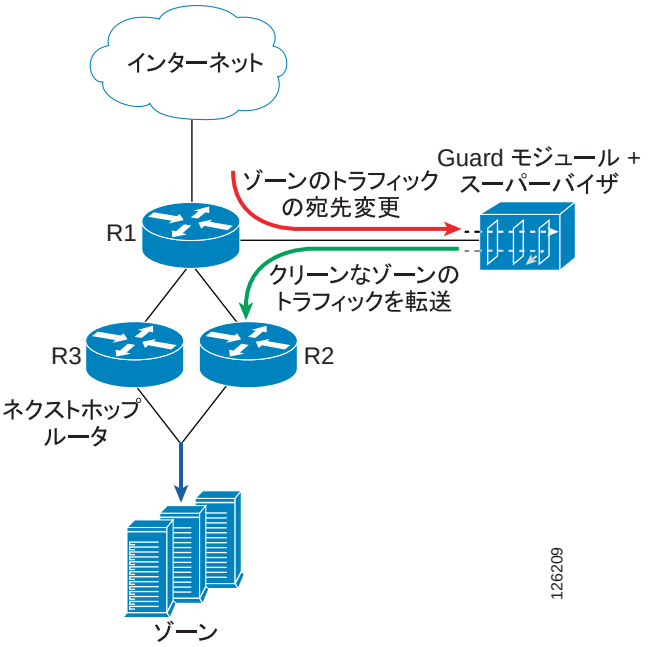
- インライン ネットワーク設定：メイン パスに存在するスイッチに Guard モジュールを設置します（つまり、ゾーン トラフィックはすでにこのスイッチを通過しています）。この設定では、Guard モジュールは、スーパーバイザ エンジンのオンボード ルーティング テーブルにスタティック ルートを追加することで、ゾーン トラフィックを宛先変更します。その後、正当なトラフィックを元の宛先に再び注入します。図 5-1 に、インライン ネットワーク設定の例を示します。

図 5-1　インライン ネットワーク設定



- アウトオブパス ネットワーク設定：ゾーン トラフィックの通常のラインにあるスイッチではなく、ラインの外側にあるスイッチに Guard モジュールを設置します。この設定では、ゾーン トラフィックはゾーンの通常のラインからスイッチに宛先変更されています。ハイジャックを設定する場合、Guard モジュールはスーパーバイザ エンジンのオンボード ルーティング テーブルにスタティック ルートを追加します。このスタティック ルートが BGP などの関連ルーティング プロトコルによってアドバタイズされるよう、ルーティング テーブルの再配布をあらかじめ設定しておく必要があります。図 5-2 に、アウトオブパス ネットワーク設定の例を示します。

図 5-2　　アウトオブパス ネットワーク設定



宛先変更のメカニズム

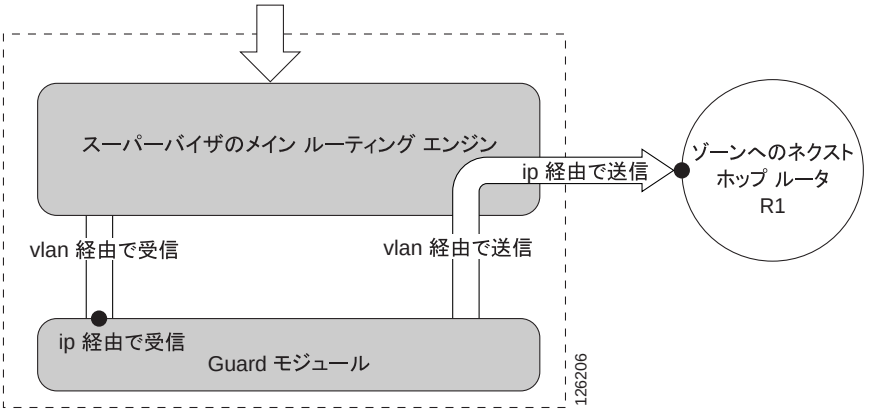
Guard モジュールの宛先変更の設定は、グローバルです。宛先変更の設定では、パケットを各サブネットにルーティングする方法を定義します。また、ハイジャックと注入の両方に必要なルートを定義します。Guard モジュールがゾーンを保護している場合、またはユーザがラーニング プロセスをアクティブにする場合、Guard モジュールは宛先変更の設定とゾーンの定義を確認し、そのゾーンを宛先とするトラフィックを宛先変更する方法と、トラフィックをゾーンのメイン トラフィック パスに再び注入する方法を判別します。

Guard モジュールは、スーパーバイザ エンジン（RHI）に対して内部プロトコルを使用し、スーパーバイザ エンジンのオンボード ルーティング テーブルにルートを追加します。Guard モジュールは、Guard モジュールがゾーンを保護してい

る場合、またはユーザがゾーンのラーニング プロセスをアクティブにする場合にルートを追加します。ゾーン保護およびラーニング プロセスが終了すると、それらのルートは削除されます。

図 5-3 に、スーパーバイザ エンジンのオンボード ルーティング テーブルと Guard モジュールの間でパケットをルーティングする方法を図示します。

図 5-3　　宛先変更のメカニズム



（注）　トラフィックのハイジャックと注入には、同じ VLAN を設定できます。

ハイジャックと注入のパラメータは、システム管理者が設定できます。

この項では、次のトピックについて取り上げます。

- ハイジャック パラメータ
- 注入パラメータ
- 宛先変更ルートの表示

ハイジャック パラメータ

ゾーンの保護をアクティブにすると、スーパーバイザ エンジンのオンボード ルーティング エンジンが、ゾーン トラフィックを Guard モジュールに宛先変更します。スーパーバイザ エンジンから Guard モジュールへのトラフィックは、VLAN *receive-via-vlan* に宛先変更されます。Guard モジュールは IP アドレス *receive-via-ip* を使用して、この VLAN でゾーン トラフィックを受信します。トラフィックのハイジャックと注入の両方には、同じ VLAN を使用できます。

Guard モジュールは、スーパーバイザ エンジンのオンボード ルーティング テーブルにスタティック ルートをインストールします。このとき、ゾーン トラフィックが Guard モジュールに確実に宛先変更されるよう、ゾーンへのネクストホップとして Guard モジュールを指します。Guard モジュールは、最長プレフィクスの照合アルゴリズムを使用します。つまり、各ルートをより長いプレフィクスを持つ 2 つのルートに分割し、これらのルートをスーパーバイザ エンジンのオンボード ルーティング テーブルにアドバタイズします。たとえば、24 ビット長のゾーン サブネット（クラス C）のルートは、25 ビット長のゾーン サブネットの 2 つのルートとして発行されます。

複数のハイジャック ルートを追加できます。各ハイジャック ルートは、ルート プリファレンスを定義する重みを持ちます。スーパーバイザ エンジンのオンボード ルーティング エンジンは、最大の重みを持つパスを優先的に使用します。デフォルトでは、すべてのハイジャック ルートに重みとして 1 が追加されています。デフォルトの重みを変更して、複数のハイジャック ルート間のプリファレンスを定義することができます。

ハイジャック パラメータを注入ルートに関連付けることができます。または、グローバルなハイジャック パラメータを設定することもできます。

（注）　ハイジャック パラメータを入力しない場合は、Guard モジュールがパラメータを動的に設定します。VLAN ID 値は、Guard モジュールのインターフェイス *giga2* に定義された VLAN ID に動的に設定され、*receive-via-ip* はその VLAN の IP アドレスに設定されます。VLAN が定義されていない場合、VLAN ID は *1* に設定され、*receive-via-ip* は *giga2* インターフェイスの IP アドレスに設定されます。

ハイジャック パラメータを注入ルートに関連付ける方法については、次の項、「注入パラメータ」を参照してください。

グローバルなハイジャック パラメータを設定するには、次のコマンドを入力します。

diversion hijacking {**receive-via-ip** *receive-via-ip* | **receive-via-vlan** *receive-via-vlan* | **weight** *weight*}

表 5-1 で、**diversion hijacking** コマンドの引数について説明します。

表 5-1　diversion hijacking コマンドの引数

パラメータ	説明
receive-via-ip	スーパーバイザ エンジンがゾーン トラフィックを転送する Guard モジュールの IP アドレス。
receive-via-vlan	スーパーバイザ エンジンがゾーン トラフィックを Guard モジュールに転送するときに使用される VLAN。
weight	宛先変更ハイジャック ルートの重み。デフォルト値は 1 です。

デフォルト値を復元する場合は、**no diversion hijacking** コマンドを使用します。

注入パラメータ

Guard モジュールは、宛先変更されたストリームから悪意のあるパケットを削除し、正当なトラフィックを、スーパーバイザ エンジンのオンボード ルーティング エンジン（レイヤ 3）に戻すか、またはゾーンのメイン トラフィック パス（レイヤ 2）に直接戻します。正当なトラフィックは VLAN *send-via-vlan* 上で送信されます。レイヤ 2 注入の場合は、ネクストホップ ルータと Guard モジュールが同じ VLAN 上に存在する必要があります。レイヤ 2 でゾーンのメイン トラフィック パスにトラフィックを注入するには、ゾーンへのネクストホップがネクストホップ ルータの IP アドレスになるように設定します。

注意　L2 注入を設定する場合は、ネクストホップ ルータとしてスーパーバイザ エンジンの IP アドレスを入力しないでください。このように入力すると、ルーティング ループが発生する場合があります。

注入パラメータを設定するには、次のコマンドを入力します。

diversion injection *ip-address ip-mask* **nexthop** *next-hop*

ip-address および *ip-mask* 引数にはゾーンの IP アドレスを指定し、*next-hop* 引数にはネクストホップ ルータの IP アドレスを指定します。

IP アドレスとサブネット マスクは、特定のゾーンのものと一致する必要はありません。これらは、ゾーン定義のサブセットにすることも、複数のゾーンのサブセットにすることもできます。たとえば、1 つまたは 2 つのコマンドを使用して、候補となる数百ものゾーンのネットワークについて宛先変更を設定することができます。

ハイジャック パラメータを注入ルートに関連付けることができます。または、グローバルなハイジャック パラメータを設定することもできます。

ハイジャック パラメータを注入ルートに関連付けるには、次のコマンドを入力します。

diversion injection *ip-address ip-mask* **nexthop** *next-hop* [**hijacking** [**receive-via-ip** *receive-via-ip*] [**receive-via-vlan** *receive-via-vlan*] [**weight** *weight*]]

ip-address および *ip-mask* 引数にはゾーンの IP アドレスを指定し、*next-hop* 引数にはネクストホップ ルータの IP アドレスを指定します。ハイジャック パラメータについては、前の項、「ハイジャック パラメータ」を参照してください。

宛先変更ルートの表示

Guard モジュールは、RHI メッセージを使用して、スーパーバイザ エンジンのオンボード ルーティング テーブルを変更します。Guard モジュールは、Guard モジュールがゾーンを保護している場合、またはユーザがゾーンのラーニング プロセスをアクティブにする場合にルートを追加します。ゾーン保護およびラーニング プロセスが終了すると、それらのルートは削除されます。

Guard モジュールの宛先変更の設定を表示するには、**show diversion** コマンドを使用します。

このメッセージは、Guard モジュールがゾーンを保護しているとき、またはゾーンがラーニング プロセス中のときに、スーパーバイザ エンジン上に表示できます。

Guard モジュールがアドバタイズしたルートを表示するには、スーパーバイザ エンジンで次のコマンドを入力します。

show anomaly-guard module *module_number* **advertised-route**

module_number 引数には、モジュールが装着されているスロットの番号を指定します。

例

```
Sup# show anomaly-guard module 9 advertised-route
RHI routes added by slot 9
    ip              mask            nexthop          vlan    weight
    --------------  --------------  ---------------  ------  ------
A   192.168.252.8   255.255.255.0   192.168.8.10     8       1
```

Guard がスタティック ルートを追加したことを確認するには、スーパーバイザ エンジンのオンボード ルーティング テーブルを表示します。

スーパーバイザ エンジンで次のコマンドを入力します。

show ip route

例

```
Sup# show ip route
C    192.168.8.0/24 is directly connected, Vlan8
S    192.168.252.8/32 [1/0] via 192.168.8.10, Vlan8
.
.
.
```

インライン ネットワーク設定

インライン ネットワーク設定では、Guard モジュールはゾーンのクリティカル パスに常駐するスイッチに設置されます。つまり、ゾーンが Guard モジュールによって保護されかどうかに関係なく、ゾーン トラフィックはこのスイッチを通過します。

この項では、次のトピックについて取り上げます。

ハイジャック

宛先変更を設定する場合、Guard モジュールは RHI メッセージを使用して、スーパーバイザ エンジンのオンボード ルーティング テーブルにルートを追加します。Guard モジュールは、ゾーン トラフィックが Guard モジュールに必ず直接転送されるよう、最も長くプレフィックスが一致するルートを追加します。次に、データ フローを分析します。宛先変更されたストリームから悪意のあるパケットをすべて削除し、注入メカニズムを使用して、正当なトラフィックを元の宛先に戻します。その結果、トラフィックは継続的に目的のゾーンに流されます。

注入

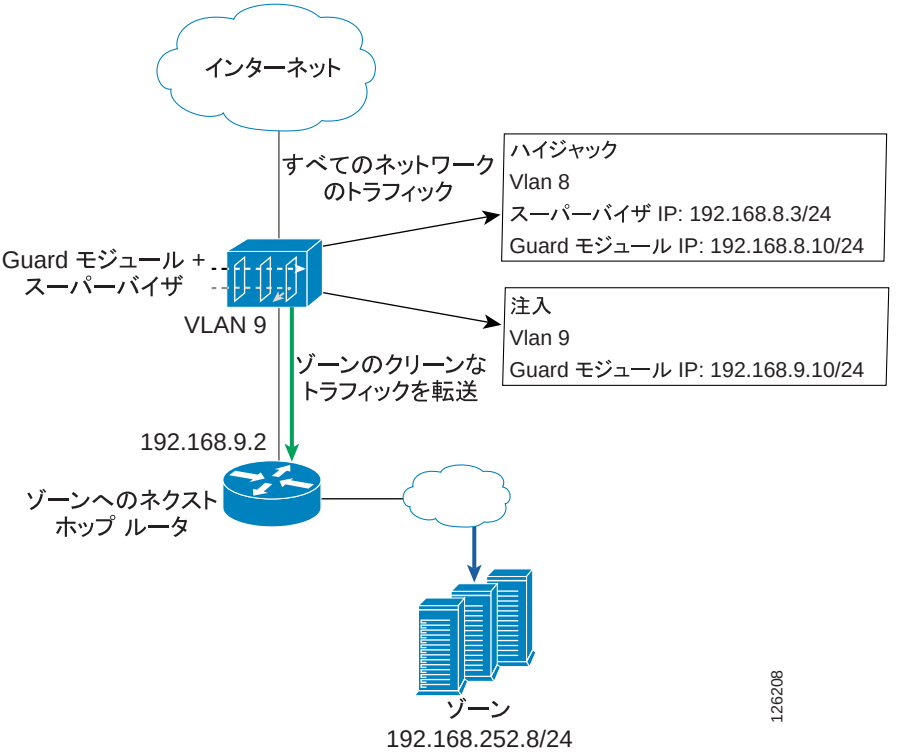
正当なトラフィックを元の宛先に戻す場合は、レイヤ 2 またはレイヤ 3 のトラフィック注入方式を使用できます。詳細については、P.5-23 の「トラフィック注入方式」を参照してください。

インラインの宛先変更の設定

図 5-4 に、インライン ネットワーク設定におけるトラフィック宛先変更の例を示します。この例では、レイヤ 3 でハイジャックが、レイヤ 2 で注入が実行されています。

（注）　宛先変更を設定する前に、ネットワークを設定する必要があります。詳細については、第 2 章「スーパーバイザ エンジンへの Guard モジュールの設定」および第 3 章「Guard モジュールの初期化」を参照してください。

図 5-4　インライン ネットワーク設定（L3 トポロジ）の例



- Guard モジュールは、スイッチのスロット 9 に設置されている。
- スイッチ上のポート GigabitEthernet2/2 は、VLAN 9 でネクストホップ ルータに接続されている。

次に、スーパーバイザ エンジンと Guard モジュールを設定する方法を、設定例とともに示します。

ステップ 1　スーパーバイザ エンジンでスイッチ インターフェイスを設定します。次のコマンドを入力します。

```
Sup# conf term
Sup(config)# vlan 8,9
Sup(config)# anomaly-guard module 9 port 2 allowed-vlan 8,9
Sup(config)# interface vlan 8
Sup(config-if)# ip address 192.168.8.3 255.255.255.0
Sup(config-if)# no ip proxy-arp
Sup(config-if)# no shutdown
Sup(config-if)# exit
Sup(config)# interface vlan 9
Sup(config-if)# ip address 192.168.9.3 255.255.255.0
Sup(config-if)# no ip proxy-arp
Sup(config-if)# no shutdown
Sup(config-if)# exit
Sup(config)# interface GigabitEthernet2/2
Sup(config-if)# switchport
Sup(config-if)# switchport mode access
Sup(config-if)# switchport access vlan 9
```

ステップ 2　Guard モジュールで Guard モジュールのインターフェイスを設定します。次のコマンドを入力します。

```
user@GUARD# conf term
user@GUARD-conf# interface giga 2
user@GUARD-conf-if-giga2# no shutdown
user@GUARD-conf-if-giga2# exit
user@GUARD-conf# interface giga 2.8
user@GUARD-conf-if-giga2.8# ip address 192.168.8.10 255.255.255.0
user@GUARD-conf-if-giga2.8# no shutdown
user@GUARD-conf-if-giga2.89# exit
user@GUARD-conf#interface giga 2.9
user@GUARD-conf-if-giga2.9# ip address 192.168.9.10 255.255.255.0
user@GUARD-conf-if-giga2.9# no shutdown
user@GUARD-conf-if-giga2.9# exit
```

ステップ 3　Guard モジュールで宛先変更を設定します。次のコマンドを入力します。

```
user@GUARD# conf term
user@GUARD-conf# diversion hijacking receive-via-ip 192.168.8.10
user@GUARD-conf# diversion hijacking receive-via-vlan 8
user@GUARD-conf# diversion injection 192.168.252.0 255.255.255.0
nexthop 192.168.9.2
```

ステップ 4　**protect** コマンドまたは **learning** コマンドを発行してゾーンをアクティブにしたら、Guard モジュールがアドバタイズしたルートと、スーパーバイザ エンジンのオンボード ルーティング テーブルに追加されたスタティック ルートを表示できます。

Guard モジュールがアドバタイズしたルートを表示するには、**show anomaly-guard module advertised-route** コマンドを使用します。スーパーバイザ エンジンのオンボード ルーティング テーブルに追加されたスタティック ルートを表示するには、**show ip route** コマンドを使用します。

次の例は、Guard モジュールがスーパーバイザ エンジンにアドバタイズしたルートを表示する方法を示しています。

```
Sup# show anomaly-guard module 9 advertised-route
RHI routes added by slot 9

    ip              mask           nexthop          vlan   weight
    --------------  -------------- ---------------  ------ ------
A   192.168.252.0   255.255.255.128192.168.8.10     8      1
A   192.168.252.128 255.255.255.128192.168.8.10     8      1
```

（注）　Guard モジュールは、Guard モジュールがゾーンを保護している場合、またはユーザがラーニング プロセスをアクティブにする場合にこれらのルートをアドバタイズします。アドバタイズされたルートを表示するには、ゾーンが現在ラーニング プロセス中であること、または Guard モジュールがゾーンを保護していることを確認します。

次の例は、スーパーバイザ エンジンのオンボード ルーティング テーブルに追加されたスタティック ルートを表示する方法を示しています。

```
Sup# show ip route
...
192.168.252.0/24 is variably subnetted, 3 subnets, 2 masks
S       192.168.252.0/25 [1/0] via 192.168.8.10, Vlan8
S       192.168.252.128/25 [1/0] via 192.168.8.10, Vlan8
```

アウトオブパス ネットワーク設定

アウトオブパス ネットワーク設定では、Guard モジュールは、ゾーン トラフィックの通常のラインにあるスイッチではなく、ラインの外側にあるスイッチに設置されます。この設定では、ゾーン トラフィックはゾーンの通常のラインからスイッチに宛先変更されています。

この項では、次のトピックについて取り上げます。

ハイジャック

宛先変更を設定する場合、Guard モジュールは RHI メッセージを使用して、スーパーバイザ エンジンのオンボード ルーティング テーブルにスタティック ルートを追加します。Guard モジュールは、ゾーン トラフィックが Guard モジュールに必ず直接転送されるよう、最も長くプレフィックスが一致するルートを追加します。BGP などの関連ルーティング プロトコルがこのスタティック ルートをアドバタイズするよう、ルーティング テーブルの配布をあらかじめ設定しておく必要があります。

Guard モジュールがゾーンを保護している場合、またはユーザがラーニング プロセスをアクティブにする場合、Guard はスーパーバイザ エンジンのオンボード ルーティング テーブルを変更します。ゾーン トラフィックが宛先変更されたルータ（宛先変更元ルータ）に BGP（EBGP または IBGP）アナウンスメントを発行するように、スーパーバイザ エンジンまたは MSFC を設定する必要があります。この BGP アナウンスメントに基づいて、宛先変更元ルータはそのルーティング テーブルを変更します。アナウンスメントにより、特定のゾーンへの最適なネクストホップとして Guard がリストされます。Guard モジュールの隣接ルータがアナウンスメントを転送しないことを保証するには、*no-advertise* と *no-export* の BGP コミュニティ ストリングを設定します。したがって、ゾーンを宛先とするパケットがネクストホップ ルータに到達すると、ルータはそのパケットをゾーンに転送します（Guard モジュールには戻しません）。

注入

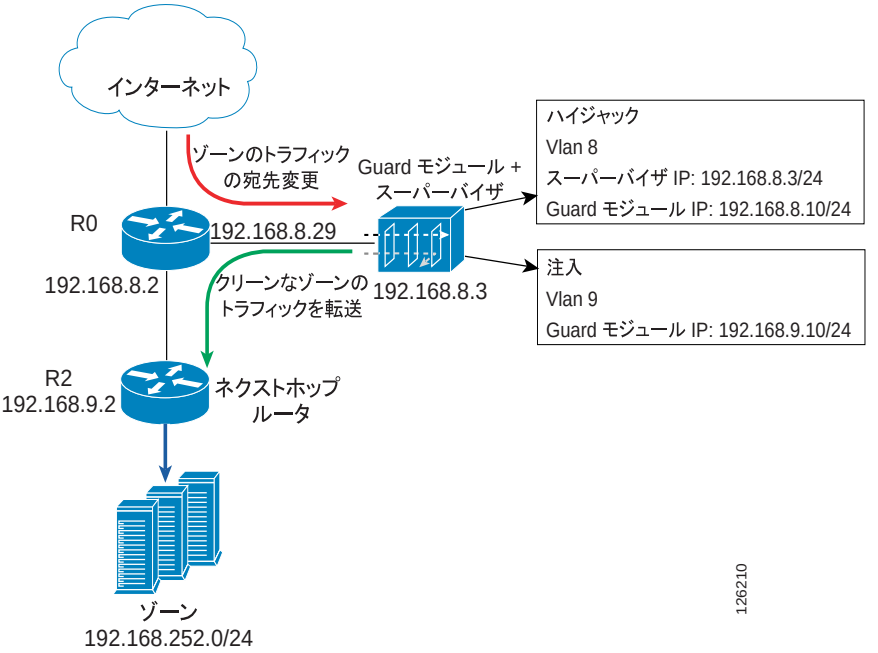
正当なトラフィックを元の宛先に戻す場合は、レイヤ 2 またはレイヤ 3 のトラフィック注入方式を使用できます。詳細については、P.5-23 の「トラフィック注入方式」を参照してください。

アウトオブパスの宛先変更の設定

図 5-4 に、アウトオブパス ネットワーク設定におけるトラフィック宛先変更の例を示します。この例では、レイヤ 3 でハイジャックが、レイヤ 2 で注入が実行されています。

（注）　宛先変更を設定する前に、ネットワークを設定する必要があります。詳細については、第 2 章「スーパーバイザ エンジンへの Guard モジュールの設定」および第 3 章「Guard モジュールの初期化」を参照してください。

図 5-5　アウトオブパス ネットワーク設定（L3 トポロジ）の例



- Guard モジュールは、スイッチのスロット 9 に設置されている。
- スイッチ上のポート GigabitEthernet2/2 は、VLAN 9 でネクストホップ ルータに接続されている。
- R0 と R2 は Autonomous System（AS; 自律システム）100 に存在し、Guard モジュールは AS 55 に存在する。

（注）　Guard モジュールがゾーンを保護している場合、トラフィックは R0 から R2 に直接流れます。ゾーン トラフィックのルートは、大きな（1 より大きい）重みを持つか、Guard モジュールが持つルートよりも限定的でないルートを持つ必要があります。

次に、スーパーバイザ エンジンと Guard モジュールを設定する方法を、設定例とともに示します。

ステップ 1　スーパーバイザ エンジンでスイッチ インターフェイスを設定します。次のコマンドを入力します。

```
sup# conf term
Sup(config)# vlan 8,9
Sup(config)# anomaly-guard module 9 port 2 allowed-vlan 8,9
Sup(config)# interface vlan 8
Sup(config-if)# ip address 192.168.8.3 255.255.255.0
Sup(config-if)# no ip proxy-arp
Sup(config-if)# no shutdown
Sup(config-if)# exit
Sup(config)# interface vlan 9
Sup(config-if)# ip address 192.168.9.3 255.255.255.0
Sup(config-if)# no ip proxy-arp
Sup(config-if)# no shutdown
Sup(config-if)# exit
Sup(config)# interface GigabitEthernet2/2
Sup(config-if)# switchport mode trunk
Sup(config-if)# switchport trunk encapsulation dot1q
Sup(config-if)# switchport
Sup(config-if)# switchport access vlan 9
Sup(config-if)# switchport mode access
```

ステップ 2　Guard モジュールがスーパーバイザ エンジンのオンボード ルーティング テーブルに追加するスタティック ルートだけが隣接ルータに発行されるよう、スーパーバイザ エンジンにルート マップを設定します。*no-advertise* と *no-export* の BGP コミュニティ ストリングを設定します。

次のコマンドを入力します。

```
sup# conf term
Sup(config)# access-list 61 permit 192.168.8.10
Sup(config)# route-map PERMIT_GUARD_ONLY permit 10
Sup(config-route-map)# match ip next-hop 61
Sup(config-route-map)# set community no-export no-advertise
Sup(config-route-map)# exit
Sup(config)# route-map PERMIT_GUARD_ONLY deny 20
```

ステップ 3　スーパーバイザ エンジンに BGP 再配布ルートを設定します。AS 100 の隣接ルータを定義します。スーパーバイザ エンジンが Guard モジュールの *receive-via-ip* アドレスに等しい宛先 IP アドレスを使用してオンボード ルーティング テーブルにスタティック ルートを追加するたびに BGP アナウンスメントを発行するように、スーパーバイザ エンジンを設定します。

次のコマンドを入力します。

```
sup# conf term
Sup(config)# router bgp 55
Sup(config-router)# bgp log-neighbor-changes
Sup(config-router)# neighbor 192.168.8.29 remote-as 100
Sup(config-router)# address-family ipv4
Sup(config-router-af)# redistribute static route-map PERMIT_GUARD_ONLY
Sup(config-router-af)# neighbor 192.168.8.29 activate
Sup(config-router-af)# no auto-summary
Sup(config-router-af)# no synchronization
Sup(config-router-af)# exit-address-family
```

ステップ 4　Guard モジュールで Guard モジュールのインターフェイスを設定します。次のコマンドを入力します。

```
user@GUARD# conf term
user@GUARD-conf# interface giga 2.8
user@GUARD-conf-if-giga2.8# ip address 192.168.8.10 255.255.255.0
user@GUARD-conf-if-giga2.8# no shutdown
user@GUARD-conf-if-giga2.8# exit
user@GUARD-conf# interface giga 2.9
user@GUARD-conf-if-giga2.9# ip address 192.168.9.10 255.255.255.0
user@GUARD-conf-if-giga2.9# no shutdown
user@GUARD-conf-if-giga2.9# exit
```

ステップ 5　Guard モジュールで宛先変更を設定します。次のコマンドを入力します。

```
user@GUARD# conf term
user@GUARD-conf# diversion hijacking receive-via-ip 192.168.8.10
user@GUARD-conf# diversion hijacking receive-via-vlan 8
user@GUARD-conf# diversion injection 192.168.252.0 255.255.255.0
nexthop 192.168.9.2
```

ステップ 6　ルータ R0 に BGP 設定値を設定します。次のコマンドを入力します。

```
RouterR0# conf term
RouterR0(config)# router bgp 100
RouterR0(config-router)# neighbor 192.168.8.3 remote-as 55
```

ステップ 7　**protect** コマンドまたは **learning** コマンドを発行してゾーンをアクティブにしたら、Guard モジュールがアドバタイズしたルートと、スーパーバイザ エンジンのオンボード ルーティング テーブルに追加されたスタティック ルートをスーパーバイザ エンジン上に表示できます。

Guard モジュールがアドバタイズしたルートを表示するには、**show anomaly-guard module advertised-route** コマンドを使用します。スーパーバイザ エンジンのオンボード ルーティング テーブルに追加されたスタティック ルートを表示するには、**show ip route** コマンドを使用します。

次の例は、Guard モジュールがアドバタイズしたルートを表示する方法を示しています。

```
Sup# show anomaly-guard module 9 advertised-route
RHI routes added by slot 9

    ip              mask            nexthop          vlan    weight
    --------------  --------------- ---------------  ------  ------
A   192.168.252.8   255.255.255.0   192.168.8.10     8       1
```

（注）　Guard モジュールは、ゾーンを保護している場合、またはユーザがラーニング プロセスをアクティブにする場合にこれらのルートをアドバタイズします。アドバタイズされたルートを表示するには、ゾーンが現在ラーニング プロセス中であること、または Guard モジュールがゾーンを保護していることを確認します。

次の例は、スーパーバイザ エンジンのオンボード ルーティング テーブルに追加されたスタティック ルートを表示する方法を示しています。

```
Sup# show ip route
...
192.168.252.0/24 is variably subnetted, 3 subnets, 2 masks
S       192.168.252.0/25 [1/0] via 192.168.8.10, Vlan8
S       192.168.252.128/25 [1/0] via 192.168.8.10, Vlan8
```

ステップ 8　ルータ R0 がゾーンへの新しいルート（Guard モジュールによってアドバタイズされたもの）をルーティング テーブルに追加したことを確認します。ルータ R0 上の BGP ルーティング テーブルを表示します。

次の例は、Guard モジュールがゾーンへの新しいルートをアドバタイズする前の BGP ルーティング テーブルを示しています。

```
RouterR0# show ip bgp
.
.
.
    Network          Next Hop       Metric LocPrf    Weight  Path
*>  192.168.252.0/24192.168.9.2    0                0       100 ?
```

次の例は、Guard モジュールがゾーンへの新しいルートをアドバタイズした後の BGP ルーティング テーブルを示しています。

```
RouterR0# show ip bgp
.
.
.
    Network          Next Hop       Metric LocPrf    Weight  Path
*>  192.168.252.0/25192.168.8.3    0                0       55 ?
*>  192.168.252.128/25192.168.8.3  0                0       55 ?

RouterR0#
```

トラフィック注入方式

この項では、Guard モジュールからネクストホップ ルータに正当なトラフィックを注入する際に使用される各方式について説明します。方式は、2 つのメイン ネットワーク トポロジによって異なります。

- レイヤ 2 トポロジ
- レイヤ 3 トポロジ

レイヤ 2 トポロジ

このトポロジでは、正当なトラフィックを元の宛先に戻すために、Guard モジュールが正当なトラフィックをネクストホップ ルータに直接転送します。スーパーバイザ エンジンがルーティングを決定する必要はありません。

Guard モジュールは、ネクストホップ ルータの IP アドレスに ARP クエリーを送信して、ネクストホップ ルータの MAC アドレスを特定します（詳細については、P.5-8 の「注入パラメータ」を参照）。次に、関連するネクストホップ ルータに接続されているスイッチ インターフェイスに正当なトラフィックを転送します。したがって、スーパーバイザ エンジンとゾーンへのネクストホップ ルータは同じ VLAN 上に存在する必要があり、Guard モジュールはその VLAN 上に IP アドレスを持っている必要があります。

設定例については、P.5-12 の「インラインの宛先変更の設定」および P.5-17 の「アウトオブバスの宛先変更の設定」を参照してください。

レイヤ 3 トポロジ

このトポロジでは、正当なトラフィックを元の宛先に再び注入するために、スーパーバイザ エンジンがルーティングを決定する必要があります。Guard モジュールは、正当なトラフィックを次の宛先のどちらかに注入できます。

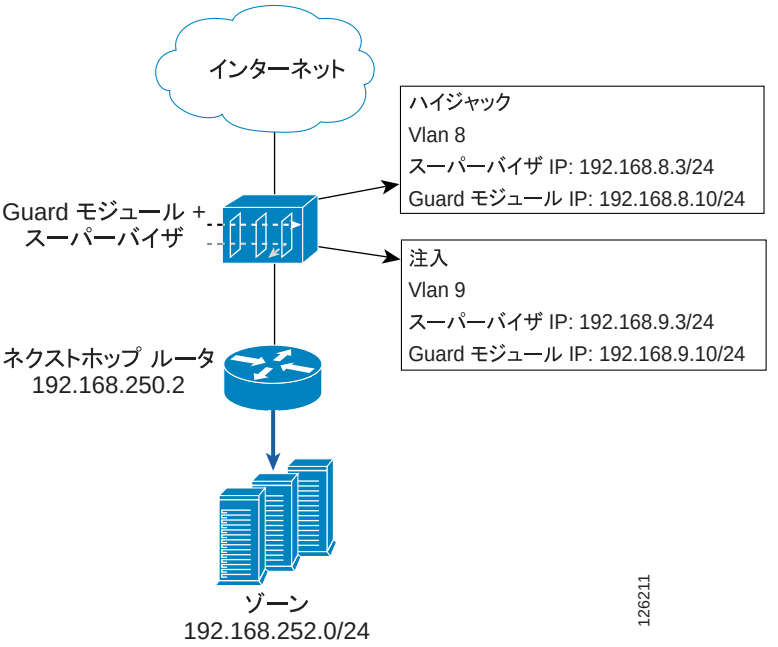
- 別のルータまたは VLAN：設定例については、P.5-12 の「インラインの宛先変更の設定」を参照してください。
- トラフィックの宛先変更元。

protect コマンドまたは **learning** コマンドを発行してゾーンをアクティブにする場合、Guard モジュールは、ゾーンへの最良のパスとしてリストされるようにルーティング テーブルを変更します（ルーティング テーブルは、ネットワーク トポロジに応じて、スーパーバイザ エンジンのオンボード ルーティング テーブルまたは隣接ルータのルーティング テーブルのどちらかです）。Guard モジュールが正当なトラフィックをトラフィックの宛先変更元に戻す場合、ルーティング ループが発生することがあります。そのようなルーティング ループが発生しないようにするには、ルーティング規則を Guard モジュールがゾーンに転送する正当なトラフィックに関連付けます。この規則は、グローバルなルーティング テーブルを上書きするように設定できます。

VPN Routing and Forwarding（VRF; VPN ルーティングおよび転送）ルーティング インスタンスを使用すると、スーパーバイザ エンジンのオンボード ルーティング テーブルに追加の転送テーブルを作成できます。この方式を使用すると、スーパーバイザ エンジンのオンボード ルーティング テーブルを使用せずにトラフィックを転送できるため、ルーティング ループを回避できます。この転送テーブルを使用して、Guard モジュールからゾーンに送信されるパケットをルーティングするための代替注入パスを定義します。転送テーブルには、ゾーンへのネクストホップ ルータにトラフィックを転送する方法についての情報だけを含めます。

ゾーン トラフィックは、ネクストホップ ルータに直接転送するか、トンネルに注入することができます。

図 5-6　　レイヤ 3 注入の設定例



この項では、次のトピックについて取り上げます。

- VRF の設定

VRF の設定

VRF は、レイヤ 3 ネットワーク トポロジで展開される注入方式です。この方式では、Guard モジュールが、正当なトラフィックをトラフィックのハイジャック元のルータに再び注入します。VRF は、インライン ネットワーク設定とアウトオブパス ネットワーク設定の両方に適用できます。

VRF を使用すると、グローバルなルーティング / 転送テーブルのほかに、もう 1 つルーティング / 転送テーブル（VRF テーブルと呼ばれる）を作成できます。このテーブルは、Guard モジュールとのインターフェイス上で受信されるトラフィックをルーティングするように設定します。

（注）　設定は、インライン ネットワーク設定とアウトオブパス ネットワーク設定の両方に適用されます。

次の 2 つのインターフェイスを設定します。

- **ハイジャック　インターフェイス**：このインターフェイスは、トラフィックを Guard モジュールに宛先変更する場合に使用します。この VLAN 上のトラフィックは、グローバル ルーティング テーブルに従って転送されます。次の例では、ハイジャック用に VLAN 8 を使用します。
- **注入インターフェイス**：このインターフェイスは、戻されたトラフィックを Guard モジュールからゾーンのメイン データ パスに注入する場合に使用します。このインターフェイスに VRF テーブルを設定します。VRF テーブル内のスタティック ルートは、Guard モジュールからゾーンに送信されたすべてのトラフィックをネクストホップ ルータに転送するように設定します。次の例では、注入用に VLAN 9 を使用します。

  トラフィックは、次の方式のどちらかを使用して注入できます。

  — 直接注入

  — トンネルを介した注入

（注）　複数のネクストホップ ルータを設定できます。

この項の設定は、図 5-6 のネットワーク設定に適用されます。Guard モジュールは、スイッチのスロット 9 に設置されています。

次に、スーパーバイザ エンジンと Guard モジュールを設定する方法を、設定例とともに示します。

ステップ 1　スーパーバイザ エンジン上に VRF テーブルを作成します。次のコマンドを入力します。

```
Sup# conf term
Sup(config)# ip vrf Guard-vrf
Sup(config-vrf)# rd 100:1
```

ステップ 2　スーパーバイザ エンジンに VRF テーブルを設定します。トラフィックの注入は、次の方式のどちらかを使用して設定できます。

- トラフィックをネクストホップ　ルータに直接注入する。詳細については、P.5-28 の「直接注入」を参照してください。
- トンネルを介してトラフィックを注入する。詳細については、P.5-29 の「トンネルを介した注入」を参照してください。

ステップ 3　スーパーバイザ エンジンに VLAN インターフェイスを設定し、このインターフェイスを Guard モジュールに関連付けます。次のコマンドを入力します。

```
Sup# conf term
Sup(config)# interface vlan 8
Sup(config-if)# ip address 192.168.8.3 255.255.255.0
Sup(config-if)# no ip proxy-arp
Sup(config-if)# no ip directed-broadcast
Sup(config-if)# no shutdown
Sup(config-if)# exit
Sup(config)# interface vlan 9
Sup(config-if)# ip vrf forwarding Guard-vrf
Sup(config-if)# ip address 192.168.9.3 255.255.255.0
Sup(config-if)# no ip proxy-arp
Sup(config-if)# no shutdown
Sup(config-if)# exit
Sup(config)# anomaly-guard module 9 port 2 allowed-vlan 8,9
```

ステップ 4　Guard モジュールで Guard モジュールのインターフェイスを設定します。次のコマンドを入力します。

```
user@GUARD# conf term
user@GUARD-conf# interface giga 2.8
user@GUARD-conf-if-giga2.8# ip address 192.168.8.10 255.255.255.0
user@GUARD-conf-if-giga2.8# no shutdown
user@GUARD-conf-if-giga2.8# exit
user@GUARD-conf# interface giga 2.9
user@GUARD-conf-if-giga2.9# ip address 192.168.9.10 255.255.255.0
user@GUARD-conf-if-giga2.9# no shutdown
user@GUARD-conf-if-giga2.9# exit
```

ステップ 5　Guard モジュールで宛先変更を設定します。次のコマンドを入力します。

```
user@GUARD# conf term
user@GUARD-conf# diversion hijacking receive-via-ip 192.168.8.10
user@GUARD-conf# diversion hijacking receive-via-vlan 8
user@GUARD-conf# diversion injection 192.168.252.0 255.255.255.0
nexthop 192.168.9.3
```

直接注入

ゾーンへのルートを指定するには、VRF テーブルにスタティック ルートを追加します。**global** キーワードは、ネクストホップ ルータへのルートがグローバル ルーティング テーブルからラーニングされることを示します。スーパーバイザ エンジンで次のコマンドを入力します。

```
Sup(config)# ip route vrf Guard-vrf 192.168.252.0 255.255.255.0
192.168.250.2 global
```

または、VRF ごとに特定のルーティング プロトコル インスタンスを定義することもできます。たとえば、**address-family ipv4 vrf** コマンドを使用すると、VRF の特定の BGP インスタンスを作成できます。

トンネルを介した注入

トンネルを介した注入を設定するには、次の手順を実行します。

ステップ 1　スーパーバイザ エンジンにトンネルを設定します。次のコマンドを入力します。

（注）　次の例では、GRE トンネルを使用します。

```
Sup# conf term
Sup(config)# interface tunnel5
Sup(config-if)# ip address 192.168.145.2 255.255.255.252
Sup(config-if)# tunnel source 192.168.8.3
Sup(config-if)# tunnel destination 192.168.7.1
```

ステップ 2　ネクストホップ ルータにトンネル側を設定します。次のコマンドを入力します。

```
router# conf term
router(config)# interface tunnel5
router(config-if)# ip address 192.168.145.1 255.255.255.252
router(config-if)# tunnel source 192.168.7.1
router(config-if)# tunnel destination 192.168.8.3
```

ステップ 3　ゾーンへのルートを指定するスーパーバイザ エンジンの VRF テーブルに、スタティック ルートを追加します。**global** キーワードは、ネクストホップ ルータへのルートがグローバル ルーティング テーブルからラーニングされることを示します。次のコマンドを入力します。

```
Sup(config)# ip route vrf Guard-vrf 192.168.252.0 255.255.255.0
192.168.145.1 global
```