CHAPTER 49

コントロール プレーン ポリシングおよびレイヤ 2 制御パケット QoS の設定

（注） CoPP は、Cisco IOS Release 12.2(50)SG 以降の Supervisor 6-E および Catalyst 4900M、Cisco IOS Release 12.2(52)X0 の Supervisor 6L-E、Cisco IOS Release 12.2(54)X0 以降の Catalyst 4948-E、Cisco IOS XE 3.1.0SG 以降の Supervisor Engine 7-E、Cisco IOS XE 3.2.0XO 以降の Supervisor Engine 7L-E でサポートされます。

この章では、コントロール プレーン ポリシング（CoPP）を使用して Catalyst 4500 シリーズ スイッチを保護する方法を説明します。この章の内容は Catalyst 4500 シリーズ スイッチに固有であり、第 52 章「ACL によるネットワーク セキュリティの設定」で説明するネットワーク セキュリティ情報や手順を補足するものです。また、次のマニュアルのネットワーク セキュリティ情報や手順の補足にもなります。

- 次の URL の『*Cisco IOS Security Configuration Guide, Cisco IOS Release 12.4*』
  http://www.cisco.com/en/US/docs/ios/security/configuration/guide/12_4/sec_12_4_book.html
- 次の URL の『*Cisco IOS Security Command Reference, Cisco IOS Release 12.4*』
  http://www.cisco.com/en/US/docs/ios/security/command/reference/sec_book.html

この章の主な内容は、次のとおりです。

（注） この章で使用するスイッチ コマンドの構文および使用方法の詳細については、次の URL で『*Cisco Catalyst 4500 Series Switch Command Reference*』と関連資料を参照してください。

http://www.cisco.com/en/US/products/hw/switches/ps4324/index.html

『*Cisco Catalyst 4500 Command Reference*』に掲載されていないコマンドについては、より詳細な Cisco IOS ライブラリを参照してください。次の URL で『*Cisco IOS Command Reference*』と関連資料を参照してください。

http://www.cisco.com/en/US/products/ps6350/index.html

コントロール プレーン ポリシングの設定

ここでは、次の内容について説明します。

コントロール プレーン ポリシングについて

（注）　Catalyst 4500 スイッチは、すべての IPv6 ファースト ホップ セキュリティ機能（DHCPv6 インスペクション/ガード、レイヤ 2 の DHCPv6 リモート ID オプション、IPv6 の完全な RA ガードなど）についてハードウェア CoPP をサポートします。ただし、外向き方向でのポリシング用に ICMP v6 パケットを VFE は照合できないため、ハードウェア CoPP は Supervisor 6-E、Supervisor 6L-E、Catalyst 4900M および Catalyst 4948-E で動作しません。

CoPP 機能は、不要なトラフィックまたは DoS トラフィックから CPU を保護し、重要なコントロール プレーンおよび管理トラフィックを優先させることにより Catalyst 4500 シリーズ スイッチのセキュリティを向上させます。分類 TCAM および QoS ポリサーは、CoPP のハードウェア サポートを提供します。

CPU が管理するトラフィックは、次の 3 つの機能コンポーネント（*プレーン*）に分割されます。

- データ プレーン
- 管理プレーン
- コントロール プレーン

CoPP を使用することで、大半の CPU 行きトラフィックを保護し、ルーティングの安定性と信頼性を確保し、パケットを確実に配信することができます。特に重要なのは、DoS 攻撃から CPU を保護するために CoPP を使用できることです。

デフォルトでは、レイヤ 2 およびレイヤ 3 コントロール プレーン パケットの選択済みセットに一致する定義済み ACL のリストを受信します。さらにこれらのパケットごとに優先するポリシング パラメータを定義し、これらの ACL の一致基準を変更できます。

次の表に、定義済み ACL を示します。

定義済みの名前付き ACL	説明
system-cpp-dot1x	MAC DA = 0180.C200.0003
system-cpp-lldp	MAC DA = 0180.C200.000E
system-cpp-mcast-cfm	MAC DA = 0100.0CCC.CCC0 - 0100.0CCC.CCC7
system-cpp-ucast-cfm	MAC DA = 0100.0CCC.CCC0
system-cpp-bpdu-range	MAC DA = 0180.C200.0000 - 0180.C200.000F

定義済みの名前付き ACL	説明
system-cpp-cdp	MAC DA = 0100.0CCC.CCCC（UDLD/DTP/VTP/Pagp）
system-cpp-sstp	MAC DA = 0100.0CCC.CCCD
system-cpp-cgmp	MAC DA = 01.00.0C.DD.DD.DD
system-cpp-hsrpv2	IP Protocol = UDP、IPDA = 224.0.0.102
system-cpp-ospf	IP Protocol = OSPF、IP DA は 224.0.0.0/24 に一致
system-cpp-igmp	IP Protocol = IGMP、IP DA は 224.0.0.0/3 に一致
system-cpp-pim	IP Protocol = PIM、IP DA は 224.0.0.0/24 に一致
system-cpp-all-systems-on-subnet	IP DA = 224.0.0.1
system-cpp-all-routers-on-subnet	IP DA = 224.0.0.2
system-cpp-ripv2	IP DA = 224.0.0.9
system-cpp-ip-mcast-linklocal	IP DA = 224.0.0.0/24
system-cpp-dhcp-cs	IP Protocol = UDP、L4SrcPort = 68、L4DstPort = 67
system-cpp-dhcp-sc	IP Protocol = UDP、L4SrcPort = 67、L4DstPort = 68
system-cpp-dhcp-ss	IP Protocol = UDP、L4SrcPort = 67、L4DstPort = 67

データ プレーンおよび管理プレーン トラフィックの場合、ポリシングするトラフィック クラスと一致するようにユーザの ACL を定義できます。

CoPP では、MQC を使用してトラフィック分類基準を定義し、分類されたトラフィックの設定可能ポリシー アクションを指定します。MQC ではクラス マップを使用して特定のトラフィック クラスに対するパケットを定義します。トラフィックを分類した後、ポリシー マップを作成し、識別されたトラフィックにポリシー アクションを強制適用できます。**control-plane global** コンフィギュレーション コマンドでは、コントロール プレーンに直接 CoPP サービス ポリシーを付加することができます。

system-cpp-policy ポリシー マップにはポリシー マップの冒頭で定義済みの順序で定義済みのクラス マップを含める必要があります。system-cpp-policy ポリシー マップを作成する最善の方法は、グローバル マクロ system-cpp を使用することです。

system-cpp-policy ポリシー マップには、コントロール プレーン トラフィックに対する定義済みクラス マップが含まれます。システムで定義されたすべての CoPP クラス マップの名前と、それらの一致 ACL にはプレフィックス system-cpp- が付いています。デフォルトでは、トラフィック クラスに対するアクションは指定されていません。CPU 行きデータ プレーンおよび管理プレーン トラフィックに一致するクラス マップを独自に定義できます。また system-cpp-policy に定義済みクラス マップを追加できます。

コントロール プレーン ポリシングの一般的な注意事項

コントロール プレーン ポリシングの注意事項は、次のとおりです。

- ポート セキュリティは、非 IP 制御パケットの CoPP の効果をキャンセルすることがあります。

  Catalyst 4500 シリーズ スイッチでの送信元 MAC ラーニングはソフトウェアで実行されますが、制御パケット（たとえば、IEEE BPDU、CDP、SSTP BPDU、GARP/）の送信元 MAC アドレスのラーニングは許可されません。潜在的に予測されない制御パケットのレートが高いことが予想されるポートにポート セキュリティを設定した場合、システムは CPU に転送する代わりにパケットのコピーを（送信元アドレスがラーニングされるまで）生成します。

Catalyst 4500 スーパーバイザ エンジンの現在のアーキテクチャでは、CPU に送信されたパケットのコピーにポリシングを適用することはできません。CPU に転送されるパケットにのみポリシングを適用できます。パケットのコピーが制御パケットと同じレートで CPU に送信されますが、制御パケットからのラーニングが許可されていないため、ポート セキュリティはトリガーされません。ポリシングは、元のパケットではなくパケットのコピーが CPU に送信されるため適用されません。

- ARP ポリシングはクラシック シリーズ スーパーバイザ エンジンまたは固定構成スイッチではサポートされません。これは、Catalyst 4900M および 4948E スイッチ、Supervisor Engine 6-E、および Supervisor Engine 6L-E でサポートされます。
- 入力 CoPP だけがサポートされます。つまり、コントロールプレーンに関連する CLI では input キーワードだけがサポートされます。
- CPU が処理するデータ プレーンおよび管理プレーン トラフィックを識別するには、ACL とクラスマップを使用します。
- CoPP ポリシー マップでサポートされるアクションはポリシングだけです。
- CoPP ポリシー ACL では log キーワードは使用できません。

デフォルト設定

CoPP はデフォルトでディセーブルです。

コントロール プレーン トラフィックの CoPP の設定

コントロール プレーン トラフィックの CoPP を設定するには、次の作業を実行します。

	コマンド	目的
ステップ 1	`Switch# config 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	`Switch(config)# qos`	（任意）QoS をグローバルにイネーブルにします。
ステップ 3	`Switch(config)#` `macro global apply system-cpp`	（任意）system-cpp-policy ポリシー マップを作成し、コントロール プレーンに付加します。
ステップ 4	`Switch(config)# policy-map system-cpp-policy` `Switch(config-pmap)# class {system-cpp-dot1x | system-cpp-bpdu-range | system-cpp-cdp | service | system-cpp-sstp | system-cpp-cgmp | system-cpp-ospf | system-cpp-igmp | system-cpp-pim | system-cpp-all-systems-on-subnet | system-cpp-all-routers-on-subnet | system-cpp-ripv2 | system-cpp-hsrpv2 | system-cpp-ip-mcast-linklocal | system-cpp-dhcp-cs | system-cpp-dhcp-sc | system-cpp-dhcp-ss}` `Switch(config-pmap-c)# police [aggregate name] rate burst [conform-action {drop | transmit}] [{exceed-action {drop | transmit}}]}`	サービス ポリシー マップで 1 つまたは複数のシステム定義のコントロール プレーン トラフィックにアクションを関連付けます。必要に応じてこのステップを繰り返します。
ステップ 5	`Switch# show policy-map system-cpp-policy`	（任意）設定を確認します。

次に、CDP パケットをポリシングする例を示します。

```
Switch# config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# qos
Switch(config)# macro global apply system-cpp
Switch(config)# policy-map system-cpp-policy
Switch(config-pmap)# class system-cpp-cdp
Switch(config-pmap-c)# police 32000 1000 conform-action transmit exceed-action drop
Switch(config-pmap-c)# end
Switch# show policy-map system-cpp-policy
  Policy Map system-cpp-policy
    Class system-cpp-dot1x
    Class system-cpp-bpdu-range
*    Class system-cpp-cdp
      police 32000 bps 1000 byte conform-action transmit exceed-action drop
    Class system-cpp-sstp
    Class system-cpp-cgmp
    Class system-cpp-ospf
    Class system-cpp-hsrpv2
    Class system-cpp-igmp
    Class system-cpp-pim
    Class system-cpp-all-systems-on-subnet
    Class system-cpp-all-routers-on-subnet
    Class system-cpp-ripv2
    Class system-cpp-ip-mcast-linklocal
    Class system-cpp-dhcp-cs
    Class system-cpp-dhcp-sc
    Class system-cpp-dhcp-ss
Switch#
```

データ プレーンおよび管理プレーン トラフィックの CoPP の設定

データ プレーンおよび管理プレーン トラフィックの CoPP を設定するには、次の作業を実行します。

	コマンド	目的
ステップ 1	`Switch(config)# qos`	（任意）QoS をグローバルにイネーブルにします。
ステップ 2	`Switch(config)# macro global apply system-cpp`	（任意）コントロール プレーンに system-cpp-policy ポリシー マップを付加します。

	コマンド	目的
ステップ 3	Switch(config)# {**ip** | **mac**} **access-list extended** {*access-list-name*} For an **ip** access list, issue Switch(config-ext-nacl)#{**permit**|**deny**} {*protocol*} **source** {*source-wildcard*} **destination** {*destination-wildcard*} For a **mac** access list, issue Switch(config-ext-macl)#{**permit**|**deny**} **source** {*source-wildcard*} **destination** {*destination-wildcard*} [**protocol-family**] OR Switch(config)# **access-list** {*access-list-name*} {**permit | deny**} {**type-code wild-mask | address mask**}	次のように、トラフィックと照合する ACL を定義します。 • **permit :** パケットが名前付き ACL をパスする条件を設定します。 • **deny** : パケットが名前付き ACL をパスしない条件を指定します。 **(注)** ほとんどの場合は、重要なトラフィックまたは重要でないトラフィックを識別する ACL を設定する必要があります。 • **type-code :** 先頭に 0x が記された 16 ビットの 16 進数（たとえば、0x6000）。802 カプセル化パケットの場合はリンク サービス アクセス ポイント（LSAP）タイプ コードを、SNAP カプセル化パケットの場合は SNAP タイプ コードを指定します。（LSAP は SAP（サービス アクセスポイント）とも呼ばれ、802 ヘッダーの DSAP（宛先サービス アクセス ポイント）フィールドおよび SSAP（送信元サービス アクセス ポイント）フィールドのタイプ コードのことです）。 • **wild-mask** : 1 のビットが type-code 引数のビットに対応する 16 ビットの 16 進数。wild-mask は、比較時に無視する type-code 引数のビットです。（DSAP/SSAP のペアのマスクでは、2 つのビットが SAP コードの識別以外の目的で使用されるため、常に 0x0101 です）。 • **address** : 4 桁の 16 進数をドットで 3 つに区切って書かれる 48 ビット トークン リング アドレス。このフィールドはベンダー コードでのフィルタリングに使用されます。 • **mask :** 4 桁の 16 進数をドットで 3 つに区切って書かれる 48 ビット トークン リング アドレス。マスクの 1 のビットはアドレスでは無視されるビットです。このフィールドはベンダー コードでのフィルタリングに使用されます。
ステップ 4	Switch(config)# **class-map** {*traffic-class-name*} Switch(config-cmap)# **match access-group** {**access-list-number | *name*** {*access-list-name*}}	パケット分類基準を定義します。クラスに関連付けられたトラフィックを指定するには、**match** 文を使用します。
ステップ 5	Switch(config-cmap)# **exit**	グローバル コンフィギュレーション モードに戻ります。

	コマンド	目的
ステップ 6	Switch(config)# **policy-map system-cpp-policy** Switch(config-pmap)# **class** {*class-map-name*} Switch(config-pmap-c)# **police** [**aggregate** *name*] *rate burst* [**conform-action** {**drop** | **transmit**}] [{**exceed-action** {**drop** | **transmit**}}]	CoPP ポリシー マップにトラフィック クラスを追加します。トラフィック クラスにアクションを関連付けるには、**police** 文を使用します。
ステップ 7	Switch(config)# **end**	特権 EXEC モードに戻ります。
ステップ 8	Switch# **show policy-map system-cpp-policy**	入力を確認します。

次の例では、送信元アドレス 10.1.1.1 および 10.1.1.2 を持つ信頼できるホストを設定し、制約を設けずに Telnet パケットをコントロール プレーンに転送する方法を示します。残りのすべての Telnet パケットは、特定のレートでポリシングされるようにします。この例では、グローバルな QoS がイネーブルなことと、system-cpp-policy ポリシー マップが作成されたことを前提とします。

```
Switch# config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# qos
Switch(config)# macro global apply system-cpp

! Allow 10.1.1.1 trusted host traffic.
Switch(config)# access-list 140 deny tcp host 10.1.1.1 any eq telnet

! Allow 10.1.1.2 trusted host traffic.
Switch(config)# access-list 140 deny tcp host 10.1.1.2 any eq telnet

! Rate limit all other Telnet traffic.
Switch(config)# access-list 140 permit tcp any any eq telnet

! Define class-map "telnet-class."
Switch(config)# class-map telnet-class
Switch(config-cmap)# match access-group 140
Switch(config-cmap)# exit

! Add the class-map "telnet-class" to "system-cpp-policy" and define the proper action
Switch(config)# policy-map system-cpp-policy
Switch(config-pmap)# class telnet-class
Switch(config-pmap-c)# police 80000 1000 conform transmit exceed drop
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit

! Verify the above configuration steps
Switch# show policy-map system-cpp-policy
  Policy Map system-cpp-policy
    Class system-cpp-dot1x
    Class system-cpp-bpdu-range
    Class system-cpp-cdp
      police 32000 bps 1000 byte conform-action transmit exceed-action drop
    Class system-cpp-sstp
    Class system-cpp-cgmp
    Class system-cpp-ospf
    Class system-cpp-hsrpv2
    Class system-cpp-igmp
    Class system-cpp-pim
    Class system-cpp-all-systems-on-subnet
```

```
    Class system-cpp-all-routers-on-subnet
    Class system-cpp-ripv2
    Class system-cpp-ip-mcast-linklocal
    Class system-cpp-dhcp-cs
    Class system-cpp-dhcp-sc
    Class system-cpp-dhcp-ss
*    Class telnet-class
      police 80000 1000 byte conform-action drop exceed-action drop
```

コントロール プレーン ポリシングの設定時の注意事項および制約事項

コントロール プレーン ポリシングを使用（または設定）するときは、次の注意事項および制約事項を考慮してください。

すべてのスーパーバイザ エンジン

CoPP を設定する場合は、次の注意事項に従ってください。

- 入力 CoPP だけがサポートされます。**input** キーワードだけがコントロール プレーン関連の CLI ではサポートされます。
- コントロール プレーン トラフィックは、CoPP を使用する場合にだけポリシングできます。ポリシー マップをインターフェイスまたは VLAN に付加するとき、コントロール プレーン トラフィックを含むポリシー マップが受け付けられても、入力インターフェイスまたは VLAN のトラフィックはポリシングできません。
- CPU が処理するデータ プレーンおよび管理プレーン トラフィックを識別するには、ACL とクラス マップを使用します。ユ[1]ーザ定義クラス マップは、CoPP の system-cpp-policy ポリシー マップに追加する必要があります。
- デフォルトの system-cpp-policy ポリシー マップはシステム定義クラス マップのアクションを定義しません（ポリシングなし）。
- system-cpp-policy でサポートされるアクションはポリシングだけです。
- データ プレーンおよび管理プレーン トラフィック クラスの定義に MAC アドレスと IP ACL の両方を使用できます。ただし、パケットが、コントロール プレーン トラフィックに対する定義済み ACL とも一致した場合、ポリシング（またはポリシングなし）アクションはコントロール プレーン クラスに作用します。その理由は、コントロール プレーン クラスがサービス ポリシーのユーザ定義クラスの上に表示されるためです。
- 超過アクション **policed-dscp-transmit** は CoPP ではサポートされません。
- CoPP ポリシー ACL では **log** キーワードを使用しないでください。代わりに、不正なパケットが到達したかどうかを確認するには、**show policy-map interface** コマンドの出力を表示するか、SPAN 機能を使用します。

Catalyst 4900M、Catalyst 4948E、Supervisor Engine 6-E、および Supervisor Engine 6L-E には適用しない

- コントロール プレーン トラフィックをポリシングするには、システム定義のクラス マップを使用します。
- システム定義クラス マップは、通常の QoS のポリシー マップでは使用できません。
- system-cpp-policy という名前のポリシー マップは CoPP 専用です。

1.

- グローバル QoS がイネーブルで、ポリシング アクションが指定されないかぎり、CoPP はイネーブルになりません。

CoPP のモニタ

サイト固有のポリシーを作成し、コントロール プレーン ポリシーの統計情報を監視し、CoPP をトラブルシューティングするには、**show policy-map control-plane** コマンドを入力できます。このコマンドは、実際に適用されるポリシーのダイナミック情報を表示します。このダイナミック情報には、レート情報と、ハードウェアおよびソフトウェアに設定したポリシーに準拠または超過するバイト数（およびパケット数）が含まれます。

show policy-map control-plane コマンドの出力は次のようになります。

```
Switch# show policy-map control-plane

Control Plane

Service-policy input: system-cpp-policy

    Class-map: system-cpp-dot1x (match-all)
      0 packets
      Match: access-group name system-cpp-dot1x

    Class-map: system-cpp-bpdu-range (match-all)
      0 packets
      Match: access-group name system-cpp-bpdu-range

*    Class-map: system-cpp-cdp (match-all)
      160 packets
      Match: access-group name system-cpp-cdp
**      police: Per-interface
        Conform: 22960 bytes Exceed: 0 bytes
*
    Class-map: system-cpp-sstp (match-all)
      0 packets
      Match: access-group name system-cpp-sstp

    Class-map: system-cpp-cgmp (match-all)
      0 packets
      Match: access-group name system-cpp-cgmp

    Class-map: system-cpp-hsrpv2 (match-all)
      0 packets
      Match: access-group name system-cpp-hsrpv2

    Class-map: system-cpp-ospf (match-all)
      0 packets
      Match: access-group name system-cpp-ospf

    Class-map: system-cpp-igmp (match-all)
      0 packets
      Match: access-group name system-cpp-igmp

    Class-map: system-cpp-pim (match-all)
      0 packets
      Match: access-group name system-cpp-pim

    Class-map: system-cpp-all-systems-on-subnet (match-all)
      0 packets
      Match: access-group name system-cpp-all-systems-on-subnet
```

```
    Class-map: system-cpp-all-routers-on-subnet (match-all)
      0 packets
      Match: access-group name system-cpp-all-routers-on-subnet

    Class-map: system-cpp-ripv2 (match-all)
      0 packets
      Match: access-group name system-cpp-ripv2

    Class-map: system-cpp-ip-mcast-linklocal (match-all)
      0 packets
      Match: access-group name system-cpp-ip-mcast-linklocal

    Class-map: system-cpp-dhcp-cs (match-all)
      83 packets
      Match: access-group name system-cpp-dhcp-cs

    Class-map: system-cpp-dhcp-sc (match-all)
      0 packets
      Match: access-group name system-cpp-dhcp-sc

    Class-map: system-cpp-dhcp-ss (match-all)
      0 packets
      Match: access-group name system-cpp-dhcp-ss

Class-map: telnet-class (match-all)
      92 packets
      Match: access-group 140
      police:
          cir 32000 bps, bc 1500 bytes
        conformed 5932 bytes; actions:
          transmit
        exceeded 0 bytes; actions:
          drop
        conformed 0000 bps, exceed 0000 bps

Class-map: class-default (match-any)
      0 packets
      Match: any
        0 packets
Switch#
```

コントロール プレーンのカウンタをクリアするには、**clear control-plane *** コマンドを実行します。

```
Switch# clear control-plane *
Switch#
```

すべての CoPP アクセス リスト情報を表示するには、**show access-lists** コマンドを実行します。

```
Switch# show access-lists
Extended IP access list system-cpp-all-routers-on-subnet
10 permit ip any host 224.0.0.2
Extended IP access list system-cpp-all-systems-on-subnet
10 permit ip any host 224.0.0.1
Extended IP access list system-cpp-dhcp-cs
10 permit udp any eq bootpc any eq bootps Extended IP access list
system-cpp-dhcp-sc
10 permit udp any eq bootps any eq bootpc Extended IP access list
system-cpp-dhcp-ss
10 permit udp any eq bootps any eq bootps Extended IP access list
system-cpp-igmp
10 permit igmp any 224.0.0.0 31.255.255.255 Extended IP access list
system-cpp-ip-mcast-linklocal
10 permit ip any 224.0.0.0 0.0.0.255 Extended IP access list
```

```
system-cpp-ospf
10 permit ospf any 224.0.0.0 0.0.0.255 Extended IP access list
system-cpp-pim
10 permit pim any 224.0.0.0 0.0.0.255 Extended IP access list
system-cpp-ripv2
10 permit ip any host 224.0.0.9
Extended MAC access list system-cpp-bpdu-range
permit any 0180.c200.0000 0000.0000.000f Extended MAC access list
system-cpp-cdp
permit any host 0100.0ccc.cccc
Extended MAC access list system-cpp-cgmp
permit any host 0100.0cdd.dddd
Extended MAC access list system-cpp-dot1x
permit any host 0180.c200.0003
system-cpp-sstp
permit any host 0100.0ccc.cccd
```

CoPP アクセス リストを 1 つだけ表示するには、**show access-lists system-cpp-cdp** コマンドを実行します。

```
Switch# show access-list system-cpp-cdp
Extended MAC access list system-cpp-cdp
permit any host 0100.0ccc.cccc
Switch#
```

レイヤ 2 制御パケット QoS の設定

レイヤ 2 制御パケット QoS では、物理ポートまたは LAN に到着した制御パケットのポリシングが行えます。

ここでは、次の内容について説明します。

レイヤ 2 制御パケット QoS の概要

パケットが CPU に到達する前に特定のポートの STP、CDP、VTP、SSTP、BPDU、EAPOL、および LLDP などの着信レイヤ 2 制御パケットをポリシングする場合があります。これは集約トラフィックが（CoPP の使用により）ポリシングを受ける前に最初の防御として役立つ可能性があります。デフォルトでは、ポリサーは入力方向のレイヤ 2 制御パケットには適用できません。これは、ユーザが重要なレイヤ 2 制御パケットを誤ってポリシングしたりドロップしたりしないようにします。

このアプローチは間違って制御パケットをポリシングするユーザを保護する一方、より深刻な問題が生じます。レイヤ 2 制御パケットのフラッディングが、DoS 攻撃が原因で、または設定ミスのためカスタマー ネットワーク内で発生するループが原因で非常に高いレートでスイッチ インターフェイスのいずれかにおいて受信された場合、CPU 使用率が急速に増加する可能性があります。これは、プロトコルのキープアライブおよびルーティング プロトコル アップデートの喪失など、悪影響をもたらす場合があります。レイヤ 2 制御パケットの QoS 機能では、入力方向のポート、VLAN、またはポート VLAN レベルでレイヤ 2 制御パケットをポリシングすることができます。

デフォルト設定

レイヤ 2 制御パケット QoS はデフォルトでディセーブルです。

レイヤ 2 制御パケット QoS のイネーブル化

レイヤ 2 制御パケット QoS をイネーブルにするには、次の作業を行います。

	コマンド	目的
ステップ 1	`Switch# configuration interface`	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	`Switch(config)# [no] qos control-packets [bpdu-range | cdp-vtp | eapol | sstp | protocol-tunnel | lldp]`	すべてまたは特定のパケット タイプの QoS をイネーブルにします。 すべてまたは特定のパケット タイプの QoS をディセーブルにするには、**no** キーワードを使用します。
ステップ 3	`Switch(config)# end`	コンフィギュレーション モードを終了します。
ステップ 4	`Switch# show run | inc qos control-packets`	設定を確認します。

表 49-1 に、この機能で影響を受けるパケットのタイプを示します。

表 49-1　パケット タイプと作用対象のアドレス範囲

機能をイネーブルにするパケットのタイプ	機能が作用するアドレス範囲
BPDU 範囲	0180.C200.0000 BPDU 0180.C200.0002 OAM, LACP 0180.C200.0003 EAPOL
CDP-VTP	0100.0CCC.CCCC
SSTP	0100.0CCC.CCCD
LLDP	0180.C200.000E

次に、CDP パケットに対して QoS をイネーブルにし、インターフェイス gi3/1 および VLAN 1 に到着する CDP パケットにポリサーを適用する例を示します。

```
Switch# config terminal
Switch(config)# qos control-packets cdp-vtp
Switch(config)# end
Switch# show run | inc qos control-packets
qos control-packets cdp-vtp
Switch# show class-map
Class Map match-any system-control-packet-cdp-vtp (id 1)

       Match access-group name system-control-packet-cdp-vtp

Create a policy map and attach it to interface gi3/1 , vlan 1
Switch# config terminal
Switch(config)# policy-map police_cdp
Switch(config-pmap)# class system-control-packet-cdp-vtp
Switch(config-pmap-c)# police 32k
Switch(config-pmap-c)# end

Switch(config)# interface gi3/1
```

```
Switch(config-if)# vlan 1
Switch(config-if-vlan-range)# service-policy in police_cdp
Switch(config-if-vlan-range)# exit
Switch(config-if)# exit
Switch(config)# exit
Switch# show policy-map interface gi3/1

GigabitEthernet3/1 vlan 1

  Service-policy input: police_cdp

    Class-map: system-control-packet-cdp-vtp (match-any)
      0 packets
      Match: access-group name system-control-packet-cdp-vtp
        0 packets
      police:
          cir 32000 bps, bc 1500 bytes
        conformed 0 packets, 0 bytes; actions:
          transmit
        exceeded 0 packets, 0 bytes; actions:
          drop
        conformed 0000 bps, exceed 0000 bps

    Class-map: class-default (match-any)
      0 packets
```

レイヤ 2 制御パケット QoS のディセーブル化

no qos control-packets コマンドでは、すべてのパケット タイプに関する QoS がディセーブルになります。

次に、QoS をすべてのパケット タイプに関してイネーブルにし、その後で CDP パケットの QoS をディセーブルにする例を示します。

```
Switch# show running-configuration | include qos control-packets
qos control-packets bpdu-range
qos control-packets cdp-vtp
qos control-packets lldp
qos control-packets eapol
qos control-packets sstp
qos control-packets protocol-tunnel
```

（注） すべての制御パケット（CDP および VTP、BPDU 範囲、SSTP、LLDP、プロトコル トンネリング）がイネーブルにされている場合、qos control-packets だけが NVGEN に格納されます。前の出力で示された個々のプロトコル名は、一部の制御パケットが設定されている場合にだけ NVGEN に格納されます。

```
Switch# config terminal
Switch(config)# no qos control-packets cdp-vtp
Switch(config)# end
Switch# show running-configuration | include qos control-packets
qos control-packets bpdu-range
qos control-packets lldp
qos control-packets sstp
qos control-packets protocol-tunnel
```

（注）　指定されたプロトコル タイプに対してこの機能を設定解除すると、そのプロトコル タイプを処理するユーザ設定ポリシーがただちに無効な状態になります。TCAM リソースを保存するには、MACL およびクラス マップ（自動生成またはユーザ定義）のほかポリシーを削除します。

（注）　インターフェイスがダウン ステートの場合、TCAM リソースは消費されません。

表 49-2 に、対応するパケット タイプの機能をイネーブルにすると作成される自動生成の MACL とクラス マップを示します。

表 49-2　パケット タイプおよび自動生成 MACL/クラス マップ

パケット タイプ	自動生成 MACL/クラス マップ
BPDU 範囲	mac access-list extended system-control-packet-bpdu-range permit any 0180.c200.0000 0000.0000.000c class-map match-any system-control-packet-bpdu-range match access-group name system-control-packet-bpdu-range
SSTP	mac access-list extended system-control-packet-sstp permit any host 0100.0ccc.cccd class-map match-any system-control-packet-sstp match access-group name system-control-packet-sstp
CDP-VTP	mac access-list extended system-control-packet-cdp-vtp permit any host 0100.0ccc.cccc class-map match-any system-control-packet-cdp-vtp match access-group name system-control-packet-cdp-vtp
EAPOL	mac access-list extended system-control-packet-eapol permit any any 0x888E class-map match-any system-control-packet-eapol match access-group name system-control-packet-eapol
LLDP	mac access-list extended system-control-packet-lldp permit any host 0180.c200.000e class-map match-any system-control-packet-lldp match access-group name system-control-packet-lldp
PROTOCOL TUNNEL	mac access-list extended system-control-packet-protocol-tunnel permit any host 0100.0ccd.cdd0 class-map match-any system-control-packet-protocol-tunnel match access-group name system-control-packet-protocol-tunnel

レイヤ 2 制御パケット QoS の設定例

CPU に対する DoS 攻撃を防ぐために CoPP およびレイヤ 2 制御パケット QoS を一緒に使用できます。次の例では、インターフェイス gi3/1、VLAN 1 および VLAN 2 に到着する BPDU は、32 kbps と 34 kbps にそれぞれ制限されます。CPU への集約 BPDU トラフィックは、CoPP を使用して 50 kbps にさらにレート制限されます。

```
Switch(config)# qos control-packets
Switch(config)# policy-map police_bpdu_1
Switch(config-pmap)# class system-control-packet-bpdu-range
Switch(config-pmap-c)# police 32k 1000
Switch(config-pmap-c-police)# exit
Switch(config-pmap-c)# exit
Switch(config-pmap)# policy-map police_bpdu_2
Switch(config-pmap)# class system-control-packet-bpdu-range
Switch(config-pmap-c)# police 34k
Switch(config-pmap-c-police)# exit
```

レイヤ 2 制御パケット QoS の設定

```
Switch(config)# interface gi3/1
Switch(config-if)# vlan-range 1
Switch(config-if-vlan-range)# service-policy in police_bpdu_1
Switch(config-if-vlan-range)# exit
Switch(config-if)# interface gi3/2
Switch(config-if)# vlan-range 2
Switch(config-if-vlan-range)# service-policy in police_bpdu_1
Switch(config-if-vlan-range)# exit
```

コントロール プレーン ポリシーの設定

```
Switch(config)# macro global apply system-cpp
Switch(config)# policy-map system-cpp-policy
Switch(config-pmap)# class system-cpp-bpdu-range
Switch(config-pmap-c)# police 50k
Switch(config-pmap-c-police)# exit
Switch(config-pmap-c)# exit
```

（注） ポリサー リソースの消費を減らすために、ポートまたは VLAN のグループに適用された名前付き集約ポリサーを使用することもできます。

（注） システムによって自動生成したクラス マップおよび MACL を変更しないでください。この処理によって、実行コンフィギュレーションがファイルから更新されたとき、またはスイッチのリロードの際に、予期しない動作が起きる可能性があります。

システム生成クラス マップまたは MACL を調整または変更するには､ユーザ定義クラス マップおよび MACL を適用します。

（注） ユーザ定義クラス マップ名はプレフィックス system-control-packet で開始する必要があります。そうしないと、特定のハードウェア（Catalyst 4924、Catalyst 4948、Catalyst 4948-10GE、Supervisor Engine II-Plus、Supervisor Engine II+10GE、Supervisor Engine V、Supervisor Engine V-10GE）は、設定された QoS アクションを実行しない可能性があります。

たとえば、次に示すのは、プレフィックス system-control-packet で始まるため、レイヤ 2 制御パケットをポリシングするための有効なユーザ定義クラス マップ名です。

```
system-control-packet-bpdu1
```

```
system-control-packet-control-packet
```

このような制限はユーザ定義 MACL（アクセス グループ）に使用する名前にはありません。

次の例は、EAPOL と BPDU パケットを識別するユーザ定義のクラス マップおよび MACL を作成する方法を示します。自動生成された system-control-packet-bpdu の範囲が 3 パケット タイプ（BPDU、EAPOL と OAM）と一致するため、このトラフィック クラスをポリシングすると 3 つすべてのパケット タイプに影響します。異なるレートで BPDU と EAPOL パケットをポリシングするには、次のようにユーザ定義 MACL およびクラス マップを設定します。

```
Switch(config)# mac access-list extended system-control-packet-bpdu
Switch(config-ext-macl)# permit any host 0180.c200.0000
Switch(config-ext-macl)# exit
Switch(config)# class-map match-any system-control-packet-bpdu
Switch(config-cmap)# match access-group name system-control-packet-bpdu
Switch(config-cmap)# exit

Switch(config)# mac access-list extended system-control-packet-eapol
Switch(config-ext-macl)# permit any host 0180.c200.0003
Switch(config-ext-macl)# exit
Switch(config)# class-map match-any system-control-packet-eapol
Switch(config-cmap)# match access-group name system-control-packet-eapol
Switch(config-cmap)# exit
```

レイヤ 2 制御パケット QoS の注意事項および制約事項

レイヤ 2 制御パケット QoS を使用（または設定）するときは、次の注意事項および制約事項を考慮してください。

- レイヤ 2 制御パケット QoS をイネーブルにすると、スイッチのすべてのポートに適用されます。レイヤ 2 制御パケットが、ポートまたは VLAN に付加するポリシーで明示的に分類されていない場合は、class-default のアクションが通常の QoS ルールに従って適用します。
- 制御パケットと一致する分類子をポリシー マップの冒頭に置き、後に他のトラフィック クラスを続けて、レイヤ 2 制御パケットが誤った QoS アクションの対象にならないようにします。
- デフォルト クラス（class-default）のアクションの適用は、スーパーバイザ エンジンのタイプによって異なります。
  - NetFlow がサポートされている Supervisor Engine V-10GE : class-default に関連付けられたアクションは、一致しない制御パケットに適用されません。デフォルトの許可アクションが適用されます。system-control-packet で始まるクラス マップに関連付けるアクションだけが制御パケットに適用されます。
  - 他のすべてのスーパーバイザ エンジン : class-default に関連付けられたアクションは、一致しない制御パケットに適用されます。
- BPDU 範囲の機能をイネーブルにすると、EAPOL パケットは最初の 802.1X 認証フェーズが完了した後で初めてポリシングされます。

IPv6 制御トラフィックのポリシング

Catalyst 4900M、Catalyst 4948E、Supervisor Engine 6-E、および Supervisor Engine 6L-E では、OSPF、PIM および MLD などの IPv6 制御パケットは、そのようなトラフィックを分類するように IPv6 ACL を設定してから、そのようなトラフィックをポリシングするための QoS ポリシーを適用することで、物理ポート、VLAN、またはコントロール プレーンでポリシングできます。

次の例は、ポートで受信される OSPFv6、PIMv6 および MLD の制御トラフィックをポリシングする方法を示します。

次に、宛先 IP v6 アドレスで OSPFv6 制御パケットを識別するようにトラフィック クラスを設定する例を示します。

```
Switch# config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# ipv6 access-list ospfv6
Switch(config-ipv6-acl)# permit ipv6 any host ff02:5
Switch(config-ipv6-acl)# exit
Switch(config)# class-map ospfv6Class
Switch(config-camp)# match access-group name ospfv6
Switch(config-camp)# exit
```

次に、宛先 IPv6 アドレスによって PIMv6 制御パケットを識別するようにトラフィック クラスを設定する例を示します。

```
Switch(config)# ipv6 access-list pimv6
Switch(config-ipv6-acl)# permit ipv6 any host ff02::d
Switch(config-ipv6-acl)# exit
Switch(config)# class-map pimv6Class
Switch(config-cmap)# match access-group name pimv6
Switch(config-cmap)# exit
```

次の例では、MLD プロトコル制御パケットを識別するようにトラフィック クラスを設定する方法を示します。

```
Switch(config)# ipv6 access-list mldv1
Switch(config-ipv6-acl)# permit icmp any any mld-query
Switch(config-ipv6-acl)# permit icmp any any mld-report
Switch(config-ipv6-acl)# permit icmp any any mld-reduction
Switch(config-ipv6-acl)# exit
Switch(config)# class-map mldClass
Switch(config-cmap)# match access-group name mldv1
Switch(config-cmap)# exit
```

次に、OSPFv6、PIMv6 および MLD トラフィック クラスをポリシングするように QoS ポリシーを設定する例を示します。

```
Switch(config)# policy-map v6_control_packet_policy
Switch(config-pmap)# class mldClass
Switch(config-pmap-c)# police 32k
Switch(config-pmap-c-police)# class ospfv6Class
Switch(config-pmap-c)# police 32k
Switch(config-pmap-c)# class pimv6Class
Switch(config-pmap-c)# police 32k
Switch(config-pmap-c-police)# exit
Switch(config-pmap-c)# exit
Switch(config-pmap)# exit
Switch(config)# exit
Switch# show policy-map

Policy Map v6_control_packet_policy
    Class mldClass
     police cir 32000 bc 1500
```

```
    conform-action transmit
    exceed-action drop
  Class ospfv6Class
   police cir 32000 bc 1500
    conform-action transmit
    exceed-action drop
  Class pimv6class
   police cir 32000 bc 1500
    conform-action transmit
    exceed-action drop
```

入力方向のインターフェイス gi2/2 に対するポリシー設定の例を次に示します。

```
Switch# config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)# interface gi2/2
Switch(config-if)# service-policy in v6_control_packet_policy
Switch(config-if)# exit
```