パケット キャプチャの設定

パケットキャプチャ設定の前提条件

パケット キャプチャは Cisco Catalyst 9200 シリーズ スイッチでサポートされています。

ここでは、パケットキャプチャの設定に関する前提条件について説明します。

組み込みパケットキャプチャ設定の前提条件

組み込みパケット キャプチャ（EPC）のソフトウェア サブシステムは、その動作で CPU とメモリ リソースを消費します。さまざまなタイプの操作を行うために十分なシステム リソースを準備する必要があります。システム リソースを使用するためのガイドラインを以下の表に示します。

表 *1 : EPC* サブシステムのシステム要件

システム リソース	要件
ハードウェア	CPU 利用率の要件は、プラットフォームによって異なります。
メモリ	パケット バッファは DRAM に保存されます。パケット バッファのサイズは、ユーザが指定します。
ディスクスペース	パケットは外部のデバイスにエクスポートできます。フラッシュ ディスクでの中間保管は必要ありません。

組み込みパケットキャプチャの設定の制約事項

- レイヤ 2 EtherChannels はサポートされません。
- VRF、管理ポート、プライベート VLAN はいずれも接続ポイントとして使用することはできません。
- 組み込みパケットキャプチャ（EPC）は、ポートチャネル、スイッチ仮想インターフェイス（SVI）、およびサブインターフェイスを含む論理ポートではサポートされません。物理ポート上でのみサポートされます。
- EPC は、シャットダウン状態の VLAN インターフェイスではサポートされません。
- ユーザがスイッチポートからルーテッドポート（レイヤ 2 からレイヤ 3）へ、またはその逆へインターフェイスを変更した場合、インターフェイスが再び起動したときに、そのキャプチャポイントを削除し、新しいファイルを作成する必要があります。キャプチャポイントの停止/開始が機能しません。
- インターフェイスの出力方向にキャプチャされたパケットは、デバイスの書き換えによって行われた変更（TTL、VLAN タグ CoS、チェックサム、および MAC アドレス、DSCP、プレシデント、UP など）が反映されないこともあります。
- パケットキャプチャの最小設定可能期間は 1 秒ですが、パケットキャプチャは少なくとも 2 秒間機能します。
- キャプチャがすでにアクティブである、または開始されている場合、キャプチャポイントパラメータを変更することはできません。
- EPC は、入力のマルチキャストパケットのみをキャプチャし、出力の複製パケットはキャプチャしません。
- 入力および出力の両方のパケットの書き換え情報はキャプチャされません。
- CPU 注入されたパケットは、コントロール プレーン パケットと見なされます。したがって、これらのタイプのパケットはインターフェイスの出力キャプチャではキャプチャされません。
- コントロールプレーンパケットは、レート制限とパフォーマンスへの影響はありません。コントロール プレーン パケットのキャプチャを制限するフィルタを使用してください。
- Control and Provisioning of Wireless Access Points（CAPWAP）などのプロトコルのデコードは、DNA Advantage でサポートされています。
- 最大 8 つのキャプチャ ポイントを定義できますが、一度にアクティブにできるのは 1 つだけです。1 つ開始するには 1 つ停止する必要があります。
- MAC フィルタは、MAC アドレスに一致しても IP パケットをキャプチャしません。これは、すべてのインターフェイス（レイヤ 2 スイッチポート、レイヤ 3 ルーテッドポート）に適用されます。

- MAC ACL は、ARP などの非 IP パケットだけに使用されます。レイヤ 3 ポートまたは SVI ではサポートされません。
- MAC フィルタは、レイヤ 3 インターフェイスとレイヤ 2 パケット（ARP）をキャプチャすることはできません。
- IPv6 ベースの ACL は VACL ではサポートされません。

パケット キャプチャについて

パケットキャプチャ機能は、オンボードのパケットキャプチャ ファシリティです。ネットワーク管理者がデバイスを出入りするかデバイスを通るパケットをキャプチャすることで、パケットをローカルで分析したり、Embedded Packet Capture（EPC）を使用するオフライン分析に向けてパケットを保存してエクスポートしたりできるようにするものです。この機能は、デバイスがネットワークの管理と操作にアクティブに参加できるようにすることによって、ネットワーク操作を簡略化します。この機能は、パケットの形式に関する情報を収集することによって、トラブルシューティングを容易にします。また、アプリケーションの分析とセキュリティも容易にします。

組み込みパケットキャプチャについて

EPC は、パケットのトレースとトラブルシューティングに役立つ組み込みシステム管理機能を提供します。この機能を使用すると、ネットワーク管理者は、シスコデバイスを出入りするか通過するデータ パケットをキャプチャできます。ネットワーク管理者は、キャプチャ バッファ サイズとタイプ（循環またはリニア）およびキャプチャする各パケットの最大バイト数を定義する場合があります。パケット キャプチャ レートは、詳細な管理制御を使用してスロットリングできます。たとえば、アクセス コントロール リストを使用してキャプチャ対象パケットをフィルタリングするオプションや、最大パケット キャプチャ レートまたはサンプリング間隔の指定などの詳細な定義を行うオプションが利用できます。

Cisco IOS XE Amsterdam 17.2.1 以前では、EPC はシャットダウン状態のインターフェイスではサポートされていません。Cisco IOS XE Amsterdam 17.2.1 以降は、EPC はシャットダウン状態のインターフェイスでサポートされます。これは、インターフェイスの起動時にパケットをキャプチャする場合に便利です。

組み込みパケット キャプチャの利点

- デバイスで IPv4 および IPv6 パケットをキャプチャでき、MAC フィルタを使用したり、MAC アドレスをマッチさせたりして、非 IP パケットもキャプチャ可能。
- パケット キャプチャ ポイントを有効にする拡張可能なインフラストラクチャキャプチャポイントは、パケットがキャプチャされ、バッファと関連付けられるトラフィック トランジット ポイントです。
- 外部ツールを使用した分析に適したパケット キャプチャ ファイル（PCAP）形式でパケット キャプチャをエクスポートする機能。

- さまざまな詳細レベルでキャプチャされたデータ パケットをデコードする方法。

パケット データ キャプチャ

パケット データ キャプチャは、バッファに格納されるデータ パケットのキャプチャです。パケット データ キャプチャは、一意の名前とパラメータを入力することによって定義します。

こうしたキャプチャでは、次のアクションを実行できます。

- インターフェイスでのキャプチャのアクティブ化。
- キャプチャ ポイントへのアクセス コントロール リスト（ACL）やクラス マップの適用。

（注） Network Based Application Recognition（NBAR）と MAC スタイルのクラス マップは、サポートされていません。

- キャプチャの破棄。
- サイズやタイプなどのバッファ ストレージ パラメータの指定。サイズの範囲は 1 ～ 100 MB です。デフォルトのバッファは線形です。もう 1 つのバッファ オプションは循環です。
- プロトコル、IP アドレス、ポート アドレスに関する情報を含む一致基準の指定。

組み込みパケット キャプチャの実装方法

次のセクションでは EPC の導入方法について説明します。

パケット データ キャプチャの管理

バッファ モードでパケット データ キャプチャを管理するには、次の手順を実行します。

手順

	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	**monitor capture** *capture-name* **access-list** *access-list-name* **例：**	アクセス リストをパケット キャプチャのコア フィルタとして指定し、モニタ キャプチャを設定します。

	コマンドまたはアクション	目的
	`Device# monitor capture mycap access-list v4acl`	
ステップ 3	**monitor capture** *capture-name* **limit duration** *seconds* 例： `Device# monitor capture mycap limit duration 1000`	モニタ キャプチャの制限を設定します。
ステップ 4	**monitor capture** *capture-name* **interface** *interface-name* **both** 例： `Device# monitor capture mycap interface GigabitEthernet 0/0/1 both`	接続ポイントおよびパケット フロー方向を指定して、モニタ キャプチャを設定します。
ステップ 5	**monitor capture** *capture-name* **buffer circular size** *bytes* 例： `Device# monitor capture mycap buffer circular size 10`	パケット データをキャプチャするようにバッファを設定します。
ステップ 6	**monitor capture** *capture-name* **start** 例： `Device# monitor capture mycap start`	トラフィック トレース ポイントでパケットデータのバッファへのキャプチャを開始します。
ステップ 7	**monitor capture** *capture-name* **stop** 例： `Device# monitor capture mycap stop`	トラフィック トレース ポイントでパケット データのキャプチャを停止します。
ステップ 8	**monitor capture** *capture-name* **export** *file-location/file-name* 例： `Device# monitor capture mycap export tftp://10.1.88.9/mycap.pcap`	分析のためにキャプチャされたデータをエクスポートします。
ステップ 9	**end** 例： `Device# end`	特権 EXEC モードに戻ります。

キャプチャされたデータのモニタリングとメンテナンス

キャプチャされたパケットデータのモニタリングとメンテナンスを行うには、次の作業を実行します。キャプチャ バッファの詳細とキャプチャ ポイントの詳細を表示します。

手順

	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	**show monitor capture** *capture-buffer-name* **buffer dump** 例： `Device# show monitor capture mycap buffer dump`	（任意）キャプチャ パケットの 16 進数ダンプおよびそのメタデータを表示します。
ステップ 3	**show monitor capture** *capture-buffer-name* **parameter** 例： `Device# show monitor capture mycap parameter`	（任意）キャプチャを指定するために使用されたコマンドのリストを表示します。
ステップ 4	**debug epc capture-point** 例： `Device# debug epc capture-point`	（任意）パケット キャプチャ ポイントのデバッグを有効にします。
ステップ 5	**debug epc provision** 例： `Device# debug epc provision`	（任意）パケット キャプチャ プロビジョニングのデバッグを有効にします。
ステップ 6	**end** 例： `Device(config)# end`	特権 EXEC モードに戻ります。

組み込みパケット キャプチャの設定例

例：パケット データ キャプチャの管理

次の例では、パケット データ キャプチャを管理する方法を示します。

```
Device> enable
Device# monitor capture mycap access-list v4acl
Device# monitor capture mycap limit duration 1000
Device# monitor capture mycap interface GigabitEthernet 0/0/1 both
Device# monitor capture mycap buffer circular size 10
Device# monitor capture mycap start
Device# monitor capture mycap export tftp://10.1.88.9/mycap.pcap
Device# monitor capture mycap stop
Device# end
```

例：キャプチャされたデータのモニタリングとメンテナンス

次の例は、ASCII 形式でパケットをダンプする方法を示しています。

```
Device# show monitor capture mycap buffer dump
Starting the packet display ........ Press Ctrl + Shift + 6 to exit

0
0000: 01005E00 00020000 0C07AC1D 080045C0 ..^...........E.
0010: 00300000 00000111 CFDC091D 0002E000 .0..............
0020: 000207C1 07C1001C 802A0000 10030AFA .........*......
0030: 1D006369 73636F00 0000091D 0001 ..example.......
1
0000: 01005E00 0002001B 2BF69280 080046C0 ..^.....+.....F.
0010: 00200000 00000102 44170000 0000E000 . ......D.......
0020: 00019404 00001700 E8FF0000 0000 ..............
2
0000: 01005E00 0002001B 2BF68680 080045C0 ..^.....+.....E.
0010: 00300000 00000111 CFDB091D 0003E000 .0..............
0020: 000207C1 07C1001C 88B50000 08030A6E ...............n
0030: 1D006369 73636F00 0000091D 0001 ..example.......
3
0000: 01005E00 000A001C 0F2EDC00 080045C0 ..^...........E.
0010: 003C0000 00000258 CE7F091D 0004E000 .<.....X........
0020: 000A0205 F3000000 00000000 00000000 ................
0030: 00000000 00D10001 000C0100 01000000 ................
0040: 000F0004 00080501 0300
```

次の例は、mycapという名前のキャプチャの設定に使用するコマンドのリストを表示する方法を示しています。

```
Device# show monitor capture mycap parameter
monitor capture mycap interface GigabitEthernet 1/0/1 both
monitor capture mycap match any
monitor capture mycap buffer size 10
monitor capture mycap limit pps 1000
```

次の例は、キャプチャ ポイントをデバッグする方法を示しています。

```
Device# debug epc capture-point
EPC capture point operations debugging is on
```

```
Device# monitor capture mycap start
*Jun 4 14:17:15.463: EPC CP: Starting the capture cap1
*Jun 4 14:17:15.463: EPC CP: (brief=3, detailed=4, dump=5) = 0
*Jun 4 14:17:15.463: EPC CP: final check before activation
*Jun 4 14:17:15.463: EPC CP: setting up c3pl infra
*Jun 4 14:17:15.463: EPC CP: Setup c3pl acl-class-policy
*Jun 4 14:17:15.463: EPC CP: Creating a class
*Jun 4 14:17:15.464: EPC CP: Creating a class : Successful
*Jun 4 14:17:15.464: EPC CP: class-map Created
*Jun 4 14:17:15.464: EPC CP: creating policy-name epc_policy_cap1
*Jun 4 14:17:15.464: EPC CP: Creating Policy epc_policy_cap1 of type 49 and client type
 21
*Jun 4 14:17:15.464: EPC CP: Storing a Policy
*Jun 4 14:17:15.464: EPC CP: calling ppm_store_policy with epc_policy
*Jun 4 14:17:15.464: EPC CP: Creating Policy : Successful
*Jun 4 14:17:15.464: EPC CP: policy-map created
*Jun 4 14:17:15.464: EPC CP: creating filter for ANY
*Jun 4 14:17:15.464: EPC CP: Adding acl to class : Successful
*Jun 4 14:17:15.464: EPC CP: Setup c3pl class to policy
*Jun 4 14:17:15.464: EPC CP: Attaching Class to Policy
*Jun 4 14:17:15.464: EPC CP: Attaching epc_class_cap1 to epc_policy_cap1
*Jun 4 14:17:15.464: EPC CP: Attaching Class to Policy : Successful
*Jun 4 14:17:15.464: EPC CP: setting up c3pl qos
*Jun 4 14:17:15.464: EPC CP: DBG> Set packet rate limit to 1000
*Jun 4 14:17:15.464: EPC CP: creating action for policy_map epc_policy_cap1 class_map
epc_class_cap1
*Jun 4 14:17:15.464: EPC CP: DBG> Set packet rate limit to 1000
*Jun 4 14:17:15.464: EPC CP: Activating Interface GigabitEthernet1/0/1 direction both
*Jun 4 14:17:15.464: EPC CP: Id attached 0
*Jun 4 14:17:15.464: EPC CP: inserting into active lists
*Jun 4 14:17:15.464: EPC CP: Id attached 0
*Jun 4 14:17:15.465: EPC CP: inserting into active lists
*Jun 4 14:17:15.465: EPC CP: Activating Vlan
*Jun 4 14:17:15.465: EPC CP: Deleting all temp interfaces
*Jun 4 14:17:15.465: %BUFCAP-6-ENABLE: Capture Point cap1 enabled.
*Jun 4 14:17:15.465: EPC CP: Active Capture 1

Device# monitor capture mycap1 stop
*Jun 4 14:17:31.963: EPC CP: Stopping the capture cap1
*Jun 4 14:17:31.963: EPC CP: Warning: unable to unbind capture cap1
*Jun 4 14:17:31.963: EPC CP: Deactivating policy-map
*Jun 4 14:17:31.963: EPC CP: Policy epc_policy_cap1
*Jun 4 14:17:31.964: EPC CP: Deactivating policy-map Successful
*Jun 4 14:17:31.964: EPC CP: removing povision feature
*Jun 4 14:17:31.964: EPC CP: Found action for policy-map epc_policy_cap1 class-map
epc_class_cap1
*Jun 4 14:17:31.964: EPC CP: cleanning up c3pl infra
*Jun 4 14:17:31.964: EPC CP: Removing Class epc_class_cap1 from Policy
*Jun 4 14:17:31.964: EPC CP: Removing Class from epc_policy_cap1
*Jun 4 14:17:31.964: EPC CP: Successfully removed
*Jun 4 14:17:31.964: EPC CP: Removing acl mac from class
*Jun 4 14:17:31.964: EPC CP: Removing acl from class : Successful
*Jun 4 14:17:31.964: EPC CP: Removing all policies
*Jun 4 14:17:31.964: EPC CP: Removing Policy epc_policy_cap1
*Jun 4 14:17:31.964: EPC CP: Removing Policy : Successful
*Jun 4 14:17:31.964: EPC CP: Removing class epc_class_cap1
*Jun 4 14:17:31.965: EPC CP: Removing class : Successful
*Jun 4 14:17:31.965: %BUFCAP-6-DISABLE: Capture Point cap1 disabled.
*Jun 4 14:17:31.965: EPC CP: Active Capture 0
```

次の例は、組み込みパケットキャプチャ（EPC）のプロビジョニングをデバッグする方法を示しています。

```
Device# debug epc provision
EPC provisionioning debugging is on

Device# monitor capture mycap start
*Jun 4 14:17:54.991: EPC PROV: No action found for policy-map epc_policy_cap1 class-map
 epc_class_cap1
*Jun 4 14:17:54.991: EPC PROV:
*Jun 4 14:17:54.991: Attempting to install service policy epc_policy_cap1
*Jun 4 14:17:54.992: EPC PROV: Attached service policy to epc idb subblock
*Jun 4 14:17:54.992: EPC PROV: Successful. Create feature object
*Jun 4 14:17:54.992: EPC PROV:
*Jun 4 14:17:54.992: Attempting to install service policy epc_policy_cap1
*Jun 4 14:17:54.992: EPC PROV: Successful. Create feature object
*Jun 4 14:17:54.992: %BUFCAP-6-ENABLE: Capture Point cap1 enabled.

Device# monitor capture mycap stop
*Jun 4 14:18:02.503: EPC PROV: Successful. Remove feature object
*Jun 4 14:18:02.504: EPC PROV: Successful. Remove feature object
*Jun 4 14:18:02.504: EPC PROV: Destroyed epc idb subblock
*Jun 4 14:18:02.504: EPC PROV: Found action for policy-map epc_policy_cap1 class-map
epc_class_cap1
*Jun 4 14:18:02.504: EPC PROV: Deleting EPC action
*Jun 4 14:18:02.504: EPC PROV: Successful. CLASS_REMOVE, policy-map epc_policy_cap1,
class epc_class_cap1
*Jun 4 14:18:02.504: %BUFCAP-6-DISABLE: Capture Point cap1 disabled.
```

パケットキャプチャ設定の機能履歴と情報

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 *2:* パケットキャプチャ設定の機能情報

機能名	リリース	機能情報
パケット キャプチャの設定	Cisco IOS XE Gibraltar 16.10.1	この機能が導入されました。
ダウン状態または管理状態のインターフェイスでの EPC の設定。	Cisco IOS XE Amsterdam 17.2.1	ダウン状態または管理ダウン状態のいずれかのインターフェイスで EPC を設定しても、インターフェイスがアップ状態に変化した後のパケットキャプチャには影響しません。