

Flexible NetFlow

Flexible NetFlow 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情報とプラットフォームサポート情報を示します。

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	機能名と説明	サポートされるプラットフォーム
Cisco IOS XE 26.2.1ea	Flexible NetFlow: Flexible NetFlow 機能のサポートが導入されました。	Cisco C9550 シリーズ スマートスイッチ
Cisco IOS XE 17.18.2	Flexible NetFlow のバイトカウンタ: Flexible NetFlow の collect パラメータのバイトカウンタが導入されました。	Cisco C9350 シリーズ スマートスイッチ
Cisco IOS XE 17.18.2	Flexible NetFlow: Flexible NetFlow 機能のサポートが導入されました。	Cisco C9610 シリーズ スマートスイッチ
Cisco IOS XE 17.18.1	Flexible NetFlow: Flexible NetFlow は、アカウントティング、ネットワークのモニタリング、およびネットワークプランニングに関する詳細な統計情報を提供するネットワークトラフィックのモニタリングおよび分析ツールです。	Cisco C9350 シリーズ スマートスイッチ

Flexible NetFlow の理解

Flexible NetFlow は、アカウントティング、ネットワークのモニタリング、およびネットワークプランニングに関する詳細な統計情報を提供するネットワークトラフィックのモニタリングおよび分析ツールです。ユーザーがさまざまなパケットフィールドに基づいてカスタムフローレコードを定義できるようにすることで、ネットワークの異常検知やセキュリティ検出を強化します。

主要概念

- **フロー**: 送信元インターフェイスに到達する、同じキー値を共有するパケットの単方向のストリーム。
- **キー**: フローを識別するために使用されるパケット内の特定のフィールド（送信元/宛先 IP アドレス、ポートなど）。

キーフィールドは、フローレコード内の **match** パラメータによって識別されます。

- **非キーフィールド**: フローに関して収集されるが、フロー自体を定義するものではない、追加の関心対象フィールド（パケットカウンタなど）。

非キーフィールドは、フローレコード内の **collect** パラメータによって識別されます。

- **フローレコード**: Flexible NetFlow が収集するデータを定義する、キーフィールドと非キーフィールドの組み合わせ。
- **フローモニター**: フローレコードとエクスポートを Flexible NetFlow のキャッシュ情報と組み合わせて、1 つのフローに関して収集するデータのサイズを定義します。
- **フローエクスポート**: Flexible NetFlow によって収集されたデータをリモートシステム（NetFlow コレクタなど）にエクスポートします。
- **フローコレクタ**: NetFlow データを受信、処理、保存、および分析します。
- **フローサンプラ**: 分析対象として選択されるパケットまたはフローの数を制限することにより、デバイスの負荷を減らします。

Flexible NetFlow の利点

次に、Flexible NetFlow の利点を示します。

- スケーラビリティ、フロー情報の集約などの、大容量フロー認識。
- セキュリティの監視と DDoS の検出および識別のための拡張されたフローインフラストラクチャ。
- フロー情報をネットワーク内の特定のサービスまたはオペレーションに適応させるパケットからの新しい情報。利用できるフロー情報はカスタマイズ可能です。
- IP アカウンティング、ボーダーゲートウェイプロトコル（BGP）ポリシーアカウンティング（Cisco C9350 シリーズスマートスイッチの場合）、永続的キャッシュなど、さまざまなアカウンティング機能を置換できる包括的な IP アカウンティング機能が含まれます。

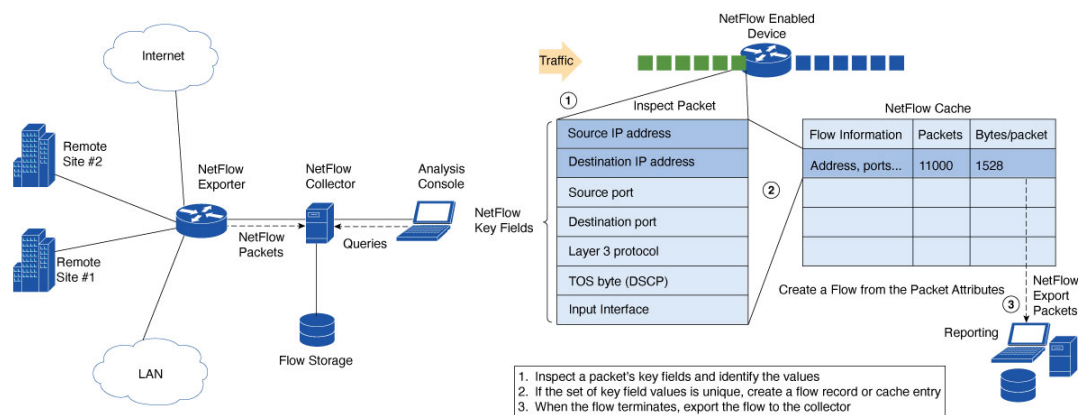
Flexible NetFlow のアプリケーション

Flexible NetFlow は、以下に示すように、さまざまなサービスに合わせてフロー情報を調整することで、ネットワークの動作を効率的に把握するために役立ちます。

- Flexible NetFlow は NetFlow をセキュリティ監視ツールとして拡張します。たとえば、ユーザーがネットワーク内で特定のタイプの攻撃を検索できるように、パケット長や MAC アドレスのために新しいフローキーを定義することができます。
- Flexible NetFlow を使用すると、パケット内のサービスクラス（CoS）に基づいて TCP アプリケーションまたは UDP アプリケーションを追跡し、ホスト間で送信されたアプリケーショントラフィックの量を迅速に識別できます。

図に、Flexible NetFlow をネットワークに導入する方法を示します。

Figure 1: Flexible NetFlow の通常の導入



Flexible NetFlow のコンポーネント

Flexible NetFlow は、トラフィック分析およびデータエクスポートに使用できるいくつかのコンポーネントで構成されます。これにより、ユーザー定義のフローレコードとコンポーネント構造が可能になり、最小限のコマンドでネットワークデバイス上のさまざまな設定を容易に実行できます。フローレコード、フローエクスポート、およびキャッシュタイプの固有の組み合わせを使用して各フローモニターを設定します。フローエクスポートの宛先 IP アドレスなどのパラメータを変更すると、フローエクスポートを使用するすべてのフローモニターが自動的に更新されます。同じフローモニターをさまざまなフローサンプラと組み合わせて使用することで、同じタイプのネットワークトラフィックを、異なるレートで、異なるインターフェイスにわたってサンプリングできます。

Flexible NetFlow のコンポーネントは、フローレコード、フローエクスポート、フローモニター、フローサンプラ、およびターゲットインターフェイスで構成され、これらが連携して、ネットワークトラフィック分析用のフローデータを定義、収集、およびエクスポートします。

フローレコード

Flexible NetFlow では、キーフィールドと非キーフィールドの組み合わせをレコードと呼びます。フローレコードは、Flexible NetFlow がパケットを識別するために使用するキーと、そのフローについて収集する追加のフィールドを指定します。キーと関連フィールドを任意の組み合わせで指定して、フローレコードを定義できます。デバイスは、幅広いキーセットをサポートします。フローレコードでは、フロー単位で収集するカウンタのタイプも定義します。64 ビットのパケットまたはバイトカウンタを設定できます。

Flexible NetFlow レコードをフローモニターに割り当てて、フローデータの格納に使用されるキャッシュを定義します。デバイスでは、フローレコードの作成時に、次のデフォルトの **match** フィールドが有効になります。

- **match datalink:** レイヤ 2 属性
- **match flow direction:** フローの方向を識別するフィールド
- **match interface:** インターフェイス属性
- **match ipv4:** IPv4 属性
- **match ipv6:** IPv6 属性

• **match transport:** トランスポート層フィールド

Flexible NetFlow のフローレコードは、2つのキーパラメータのセット（**match パラメータ**と **collect パラメータ**）を使用して定義されるユーザー定義のカスタムテンプレートです。これらのレコードは、ネットワークトラフィックフローについて収集する情報の特定のフィールドを定義し、パラメータは、ネットワークトラフィックフローに関してモニターおよびエクスポートされるフィールドを決定します。

ユーザー定義レコード

Flexible NetFlow フローモニターキャッシュに対してユーザーが定義するレコードは、「ユーザー定義レコード」と呼ばれます。Flexible NetFlow では、ユーザーがフローモニターキャッシュ用の独自のレコードを定義できます。キーフィールドと非キーフィールドを指定して、ニーズに合わせてデータ収集をカスタマイズします。

非キーフィールドは、フロー内のトラフィックに関する追加情報を提供します。**nonkey** フィールドの値の変更によって新しいフローが作成されることはありません。ほとんどの場合、**nonkey** フィールドの値はフロー内の最初のパケットからのみ取得されます。Flexible NetFlow では、パケットなどのカウンタ値を非キーフィールドとしてキャプチャできます。

ユーザー定義レコードは、QoS および帯域幅監視、アプリケーションとユーザーのトラフィックプロファイリング、DDoS 攻撃に対するセキュリティ監視などのアプリケーション用に作成できます。Flexible NetFlow のユーザー定義レコードでは、ユーザーが設定可能なサイズのパケットの連続するセクションを監視する機能を利用でき、**key** フィールドまたは **nonkey** フィールドとしてパケットのその他のフィールドや属性とともにフローレコード内で使用します。セクションにはパケットのレイヤ3 データが含まれる場合があります。

パケットのセクションフィールドでは、ユーザーが Flexible NetFlow の事前定義済みレコードの対象外のパケットフィールドを監視できます。パケットフィールドの分析機能によって、さらに詳細なトラフィック監視が可能になるため、DDoS 攻撃の調査に役立ち、URL 監視など他のセキュリティアプリケーションの実装が容易になります。

match パラメータ

match パラメータは、フローを一意に識別するフィールドを定義します。同じ **match** フィールドの値を持つパケットのみが同じフローにグループ化されます。フローレコードごとに **match** パラメータを 1 つ以上設定します。

次の表で、Flexible NetFlow の **match** パラメータについて説明します。

Table 1: **match** パラメータ

コマンド	目的
match datalink {dot1q ethertype mac vlan }	データ リンクまたはレイヤ 2 フィールドとの一致を指定します。 • dot1q: dot1q フィールドと一致します。 • ethertype: パケットの ethertype と一致します。 • mac: 送信元または宛先の MAC フィールドと一致します。 • vlan: パケットが配置される VLAN と一致します（入力または出力）。

コマンド	目的
match flow direction	フローを識別するフィールドとの一致を指定します。
match interface {input output}	インターフェイス フィールドとの一致を指定します。  Note 入力 NetFlow は出力インターフェイスと一致できず、出力 NetFlow は入力インターフェイスと一致できません。 • input: 入力インターフェイスと一致します。 • output: 出力インターフェイスと一致します。
match ipv4 {destination protocol source tos ttl version}	IPv4 フィールドとの一致を指定します。 • destination: IPv4 宛先アドレスベースのフィールドと一致します。 • protocol: IPv4 プロトコルと一致します。 • source: IPv4 送信元アドレスベースのフィールドと一致します。 • tos: IPv4 タイプオブサービス フィールドと一致します。 • ttl: IPv4 存続時間フィールドと一致します。 • version: IPv4 ヘッダーの IP バージョンと一致します。
match ipv6 {destination hop-limit protocol source traffic-class version }	IPv6 フィールドとの一致を指定します。 • destination: IPv6 宛先アドレスベースのフィールドと一致します。 • hop-limit: IPv6 ホップリミットフィールドと一致します。 • protocol: IPv6 ペイロードプロトコルフィールドと一致します。 • source: IPv6 送信元アドレスベースのフィールドと一致します。 • traffic-class: IPv6 トラフィッククラスと一致します。 • version: IPv6 ヘッダーの IP バージョンと一致します。

コマンド	目的
match transport {destination-port igmp icmp source-port}	トランスポート層フィールドとの一致を指定します。 • destination-port: 宛て先ポートと一致します。 • icmp: ICMP IPv4 および IPv6 フィールドを含む ICMP フィールドと一致します。 • igmp: IGMP フィールドと一致します。 • source-port: 転送元ポートと一致します。
match routing vrf input	着信パケットの VRF ルーティング属性との一致を指定します。

collect パラメータ

collect パラメータは、フローに関する追加情報を提供するオプションのフィールドです。これらのフィールドは、収集され、エクスポートされるフローレコードに含まれますが、フローのグループ化には影響しません。

次の表で、Flexible NetFlow の collect パラメータについて説明します。

Table 2: collect パラメータ

コマンド	目的
collect counter {bytes {layer2 {long} long} packets {long} }	カウンタ フィールドの合計バイト数と合計パケット数を収集します。
collect timestamp absolute {first last}	最初のパケットが確認された絶対時間、または最新のパケットが最後に確認された絶対時間のフィールドを収集します（ミリ秒）。

コマンド	目的
collect transport tcp flags	トランスポート TCP フラグを収集します。 • ack: TCP 確認応答フラグ • cwr: TCP 輻輳ウィンドウ縮小フラグ • ece: TCP ECN エコーフラグ • fin: TCP 終了フラグ • psh: TCP プッシュフラグ • rst: TCP リセットフラグ • syn: TCP 同期フラグ • urg: TCP 緊急フラグ  Note デバイスで、トランスポート TCP フラグを収集するように指定すると、すべての TCP フラグが収集されます。

フローエクスポート

フローエクスポートでは、フロー モニタ キャッシュ内のデータをリモートシステム（たとえば、分析および保管のために NetFlow コレクタを実行するサーバ）にエクスポートします。フローエクスポートは、設定内で個別のエンティティとして作成され、フローモニターに割り当てられて、データエクスポート機能を提供します。複数のフローエクスポートを作成して、1つまたは複数のフローモニターに適用すると、いくつかのエクスポート先を指定することができます。1つのフローエクスポートを作成し、いくつかのフローモニターに適用することができます。

Flexible NetFlow は、バージョン 9 およびバージョン 10（IPFIX）のエクスポートフォーマットをサポートしています。エクスポートプロトコルが設定されていない場合は、バージョン 9 のエクスポートフォーマットがデフォルトで適用されます。

NetFlow データ エクスポート フォーマットのバージョン 9

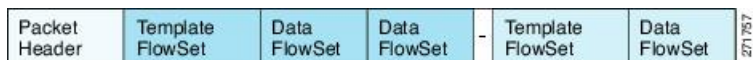
NetFlow は、主に出力としてフローレコードを生成します。NetFlow が改良され、フロー レコードのいくつかのフォーマットが向上しました。NetFlow Version 9 エクスポート フォーマットの識別機能は、テンプレートがベースとなります。テンプレートを使用すると、レコードのフォーマットを柔軟に設計できます。この柔軟性により、基本的なフローレコードのフォーマットを同時に更新することなく、将来 NetFlow サービスを改善できます。テンプレートを使用すると、次のいくつかの利点があります。

- NetFlow 向けのアプリケーションを作成するサードパーティ ビジネス パートナーは、新機能を追加するたびにアプリケーションを再コンパイルする必要はありません。代わりに、既知のテンプレートフォーマットを記述する外部のデータファイルを使用することができます。
- 新規機能は、現在の導入環境を損ねることなく、NetFlow に迅速に追加できます。

- バージョン9フォーマットは新しいプロトコルや開発中のプロトコルに適応できるため、NetFlow はこれらのプロトコルに対して「将来的に対応」します。

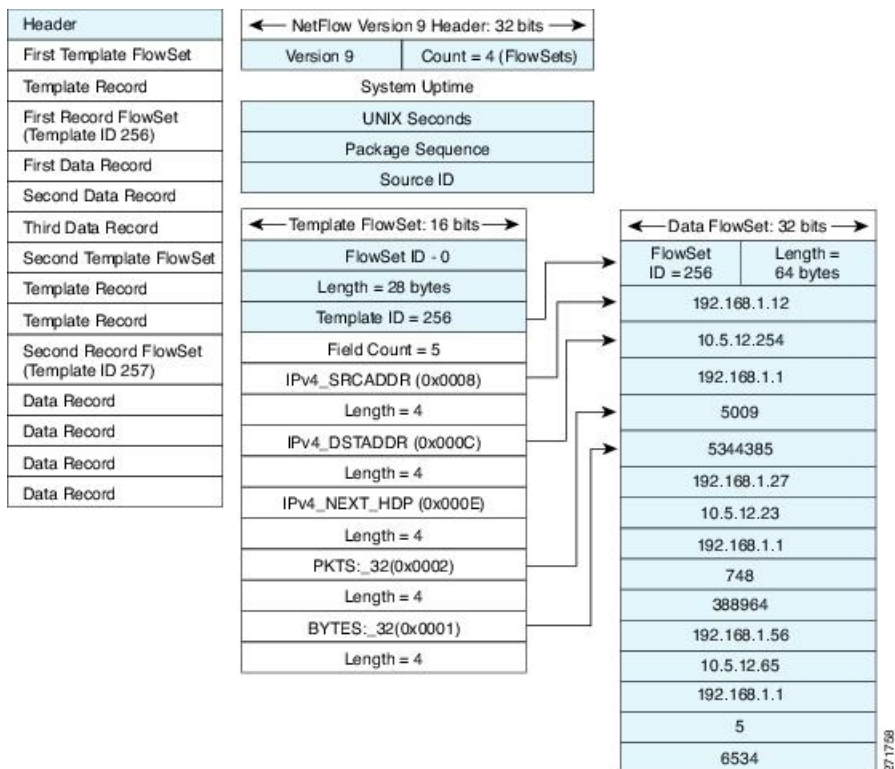
バージョン9のエクスポートフォーマットにはパケットヘッダーが含まれ、その後に1つ以上のテンプレートフローまたはデータフローのセットが続きます。テンプレートフローセットでは、将来のデータフローセットに表示されるフィールドの説明が提供されます。このようなデータフローセットは、後で同じエクスポートパケットまたは後続のエクスポートパケットで発生する可能性があります。テンプレートフローセットおよびデータフローセットは、次の図に示すように、単一のエクスポートパケットに混在させることができます。

Figure 2: バージョン9 エクスポートパケット



NetFlow バージョン9 では、送信されるデータを NetFlow コレクタが理解できるように、テンプレートデータを定期的にエクスポートします。また、テンプレートのデータフローセットもエクスポートします。Flexible NetFlow の主な利点は、ユーザーがフローレコードを設定すると、バージョン9 テンプレートに効率的に変換され、コレクタに転送されることです。次の図に、ヘッダー、テンプレートフローセットおよびデータフローセットを含めて、NetFlow バージョン9 エクスポートフォーマットの詳細な例を示します。

Figure 3: NetFlow バージョン9 エクスポートフォーマットの詳細例



NetFlow データ エクスポート フォーマット バージョン 10 (IPFIX)

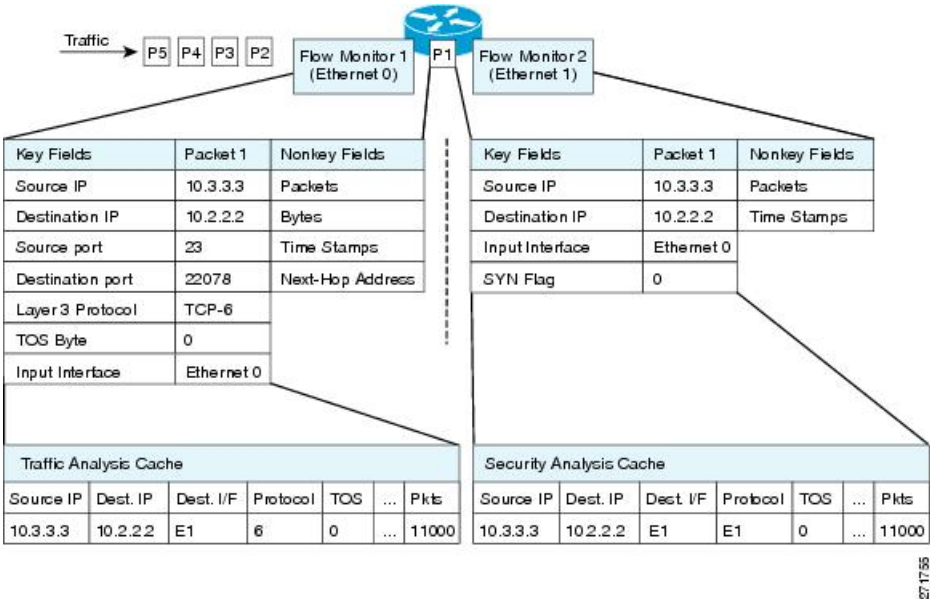
Internet Protocol Flow Information Export (IPFIX)、つまりバージョン 10 は、ユーザー定義のフローレコードを収集し、エクスポートするエクスポートプロトコルです。IPFIX は、NetFlow バージョン 9 に基づく IETF 標準 (RFC 5153 および 7011) です。IPFIX 形式は NetFlow バージョン 9 として、個別のテンプレートとレコードについて同じ原則を保ちます。IPFIX エクスポートプロトコルでは、デフォルトの宛先ポートは 4739、DSCP 値は 0、TTL は 255 です。

フロー モニタ

フローモニターは Flexible NetFlow のコンポーネントであり、ネットワークトラフィックをモニターするために、インターフェイスに適用されます。システムは、ネットワークトラフィックからフローデータを収集し、フローレコードのフィールドに基づいてフローモニターキャッシュに保存します。デバイスは、最大 8 つの機能プロファイルをサポートします。これらは、キャッシュサイズや非アクティブタイムアウトなどのフローモニターのパラメータによって定義されます。

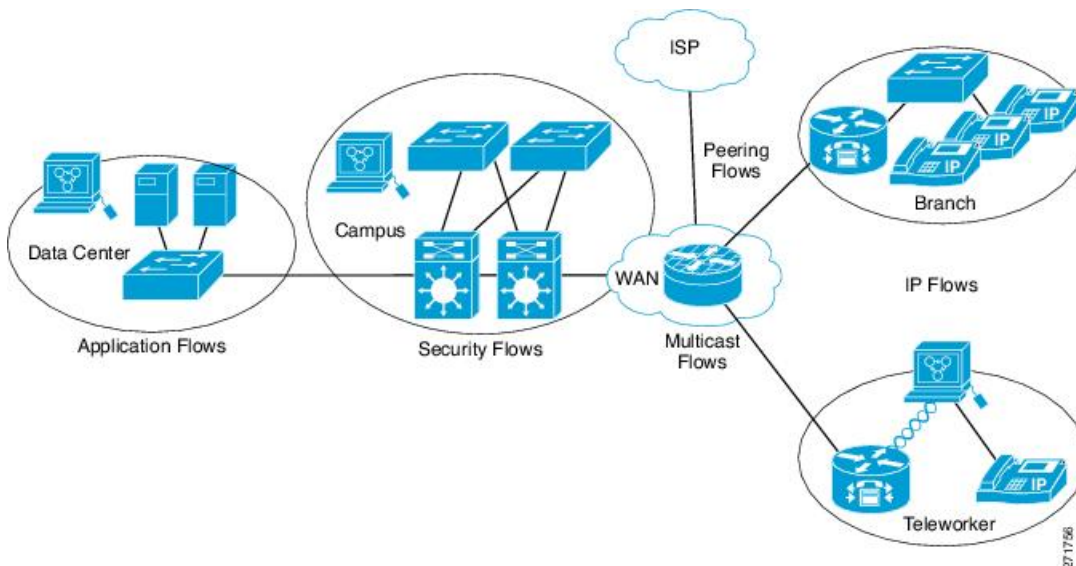
Flexible NetFlow では、同一のデータに対してさまざまなタイプのトラフィック分析が可能です。次の図は、パケット 1 が、入力インターフェイスでは標準的なトラフィック分析レコードを使用して、出力インターフェイスではセキュリティ分析レコードを使用して、どのように分析されるかを示しています。

Figure 4: 2 つのフロー モニターを使用した同じトラフィックの分析例



次の図は、カスタムレコードを使用してさまざまなタイプのフローモニターを適用するより複雑な方法の例を示しています。

Figure 5: 複雑な例: カスタムレコードを使用したさまざまなフローモニター



デフォルトのキャッシュタイプは「normal」（標準）であり、非アクティブタイムアウト時のみエントリが期限切れになります。アクティブタイムアウト時には、collect フィールドの統計のみが更新されます。キャッシュエントリは、期限切れになるとキャッシュから削除され、設定されているすべてのエクスポートを介してエクスポートされます。

フローサンプリング

フローサンプリングは、デバイス設定で作成される個別のコンポーネントです。フローサンプリングは、分析用に選択されるパケットまたはフローの数を制限することで、Flexible NetFlow を実行しているデバイス上の負荷を減らします。

フローサンプリングにより、モニタリングの精度が低下しますが、デバイスのパフォーマンスが向上します。サンプリングをフローモニターに適用すると、フローモニターが分析する必要のあるパケットまたはフローの数が減少するため、デバイスでフローモニターを実行するためのオーバーヘッド負荷が減ります。フローモニターで分析されるパケットまたはフローの数が減少すると、フローモニターのキャッシュ内の情報の精度が低下します。

ターゲットインターフェイス

ターゲットインターフェイスとは、NetFlow を適用できるインターフェイスを指します。ターゲットインターフェイスは、トラフィック統計を収集、分析、およびエクスポートするために NetFlow をセットアップする場合の設定プロセスの一部です。

レイヤ 2、IPv4、および IPv6 のトラフィックタイプが、複数のユーザー定義キャッシュ（フローモニター）としてサポートされています。特定のインターフェイスおよび方向に対して、異なるトラフィックタイプの複数のフローモニターを適用できますが、特定のインターフェイスおよび方向に対して、同じトラフィックタイプの複数のフローモニターを適用することはできません。

ユニキャスト、マルチキャスト（Cisco C9350 シリーズ スマートスイッチの場合）、およびブロードキャストのストリームは、入力方向と出力方向の両方でサポートされています。

レイヤ 2 とレイヤ 3 の両方の物理インターフェイスと、次の論理インターフェイスがサポートされています。

- ポートチャネル インターフェイス（L2 および/または L3）

- サブインターフェイス
 - 物理サブインターフェイス
 - ポートチャネル サブインターフェイス
- SVI（スイッチ仮想インターフェイス：インターフェイス VLAN）
- VLAN ID（VLAN 設定）

サポートされている Flexible NetFlow フィールド

次の表に、さまざまなトラフィックの種類と方向でサポートされている Flexible NetFlow（FNF）のフィールドを示します。

Flexible NetFlow は、「5 タプルフロー」と呼ばれる一連のキーフィールドもサポートしています。これらのフィールドは単方向の TCP および UDP セッションを表し、IPv4 および IPv6 の送信元アドレスと宛先アドレス、レイヤ 4 プロトコル、送信元ポート、および宛先ポートが含まれます。

Note

パケット内に VLAN フィールドが存在する場合、そのフィールドの長さは考慮されません。

フィールド	レイヤ 2 In	レイヤ 2 Out	IPv4 In	IPv4 Out	IPv6 In	IPv6 Out	注意
キーフィールドまたは collect フィールド：共通							
Interface input	対応	—	対応	—	対応	—	フローモニターを入力方向に適用する場合は、 match キーワードを使用し、入力インターフェイスをキーフィールドとして使用します。
Interface output	—	対応	—	対応	—	対応	フローモニターを出力方向に適用する場合は、 match キーワードを使用し、出力インターフェイスをキーフィールドとして使用します。

フィールド	レイヤ 2 In	レイヤ 2 Out	IPv4 In	IPv4 Out	IPv6 In	IPv6 Out	注意
Flow direction	対応	対応	対応	対応	対応	対応	
キーフィールド：データリンク							

フィールド	レイヤ 2 In	レイヤ 2 Out	IPv4 In	IPv4 Out	IPv6 In	IPv6 Out	注意
Ethertype	対応	対応	—	—	—	—	スイッチポートでのみサポートされています。
VLAN input	対応	—	—	—	—	—	スイッチポートでのみサポートされています。
VLAN output	—	対応	—	—	—	—	スイッチポートでのみサポートされています。
dot1q VLAN input	対応	—	—	—	—	—	スイッチポートでのみサポートされています。
dot1q VLAN output	—	対応	—	—	—	—	スイッチポートでのみサポートされています。
dot1q priority	対応	対応	—	—	—	—	スイッチポートでのみサポートされています。
MAC source address input	対応	対応	—	—	—	—	スイッチポートでのみサポートされています。
MAC source address output	—	—	—	—	—	—	
MAC destination address input	対応	—	—	—	—	—	スイッチポートでのみサポートされています。

フィールド	レイヤ 2 In	レイヤ 2 Out	IPv4 In	IPv4 Out	IPv6 In	IPv6 Out	注意
MAC destination address output	—	対応	—	—	—	—	スイッチポートでのみサポートされています。
キーフィールド: IPv4							
IPv4 version	—	—	対応	対応	—	—	
IPv4 TOS	—	—	対応	対応	—	—	
IPv4 protocol	—	—	対応	対応	—	—	送信元/宛先ポート、ICMP コード/タイプ、IGMP タイプ、TCP フラグのいずれかが使用されている場合に使用する必要があります。
IPv4 TTL	—	—	対応	対応	—	—	
IPv4 source address	—	—	対応	対応	—	—	
IPv4 destination address	—	—	対応	対応	—	—	
ICMP IPv4 type	—	—	対応	対応	—	—	
ICMP IPv4 code	—	—	対応	対応	—	—	
IGMP type	—	—	対応	対応	—	—	
フィールド	レイヤ 2 In	レイヤ 2 Out	IPv4 In	IPv4 Out	IPv6 In	IPv6 Out	注意
キーフィールド: IPv6							
IPv6 version	—	—	—	—	対応	対応	IP バージョンと同じです。

フィールド	レイヤ 2 In	レイヤ 2 Out	IPv4 In	IPv4 Out	IPv6 In	IPv6 Out	注意
IPv6 protocol	—	—	—	—	対応	対応	IP プロトコルと同じです。送信元/宛先ポート、ICMP コード/タイプ、IGMP タイプ、TCP フラグのいずれかが使用されている場合に使用する必要があります。
IPv6 source address	—	—	—	—	対応	対応	
IPv6 destination address	—	—	—	—	対応	対応	
IPv6 traffic-class	—	—	—	—	対応	対応	IP TOS と同じです。
IPv6 hop-limit	—	—	—	—	対応	対応	IP TTL と同じです。
ICMP IPv6 type	—	—	—	—	対応	対応	
ICMP IPv6 code	—	—	—	—	対応	対応	

フィールド	レイヤ 2 In	レイヤ 2 Out	IPv4 In	IPv4 Out	IPv6 In	IPv6 Out	注意
キーフィールド: トランスポート共通							
routing vrf input	—	—	対応	対応	対応	対応	
Source port	—	—	対応	対応	対応	対応	
Destination port	—	—	対応	対応	対応	対応	

フィールド	レイヤ 2 In	レイヤ 2 Out	IPv4 In	IPv4 Out	IPv6 In	IPv6 Out	注意
collect フィールド							

フィールド	レイヤ 2 In	レイヤ 2 Out	IPv4 In	IPv4 Out	IPv6 In	IPv6 Out	注意
Bytes long	対応	対応	対応	対応	対応	対応	bytes long と bytes layer2 long では同じ出力が表示されます。
Packets long	対応	対応	対応	対応	対応	対応	
Timestamp absolute first	対応	対応	対応	対応	対応	対応	
Timestamp absolute last	対応	対応	対応	対応	対応	対応	
TCP flags	対応	対応	対応	対応	対応	対応	すべてのフラグを収集します。
Bytes layer2 long	対応	対応	対応	対応	対応	対応	bytes long と bytes layer2 long では同じ出力が表示されます。

Note

tcp flags については、レイヤ 2 入力およびレイヤ 2 出力フィールドは Cisco C9610 シリーズ スマートスイッチではサポートされていません。

デフォルト設定

次の表は、デバイスに対する Flexible NetFlow のデフォルト設定を示します。

Table 3: デフォルトの *Flexible NetFlow* 設定

設定	デフォルト
フロー アクティブ タイムアウト	1800 秒
フローの非アクティブタイムアウト	15 秒

Flexible NetFlow がポートチャネルで設定されており、いずれかのメンバーポートで障害が発生すると、フローが他のアクティブなメンバーポートに一時的に移動するため、非アクティブタイムアウトの期限が切れるまで追加のフローが表示されます。このフローの一時的な重複は、フローが他のメンバーに移動され、非アクティブなフローがタイムアウトになるまでエクスポートされつづけるために発生します。この動作は、Doppler SVL や MEC 設定などのマルチ ASIC シナリオでも見られます。

Note

これにより、機能への影響や、フローモニタリングの精度への影響はありません。

サポートされている Flexible NetFlow の機能

ここでは、Flexible NetFlow でサポートされている機能について説明します。

VLAN 上でのブリッジ型 NetFlow

VLAN 上のブリッジ型 NetFlow を使用すると、VLAN 内でブリッジされるトラフィックの NetFlow データのモニタリングと収集が可能になります。レイヤ 3 インターフェイス間でルーティングされるトラフィックをキャプチャする従来のルーティング型 NetFlow とは異なり、ブリッジ型 NetFlow は、VLAN 内のレイヤ 2 でトラフィックをキャプチャします。そのため、トラフィックが VLAN から出ないもののモニターする必要がある環境で役立ちます。

この機能は、キャンパスネットワーク、データセンター、仮想化環境など、同じ VLAN またはサブネット内のデバイス間で大量のトラフィックが存在するネットワークで特に役立ちます。

Flexible NetFlow の入力および出力 VRF

Flexible NetFlow の入力 VRF 機能を使用して、キーフィールドとして Virtual Routing and Forwarding (VRF) ID を収集するフローレコードがある入力フローモニターを適用することにより、デバイスで着信パケットから VRF ID を収集できます。

Flexible NetFlow の出力 VRF 機能を使用して、キーフィールドとして VRF ID を収集するフローレコードがある出力フローモニターを適用することにより、デバイスで発信パケットから VRF ID を収集できます。

Note

出力の VRF ID match フィールドは、Cisco C9610 シリーズ スマートスイッチではサポートされていません。

レイヤ 2 およびレイヤ 3 Flexible NetFlow

Flexible Netflow レイヤ 2 機能を使用すると、トラフィックから MAC アドレスや VLAN ID などのレイヤ 2 フィールドの統計を収集できます。この機能は、ローカルネットワークまたはブロードキャストドメイン内のトラフィックのモニタリングに焦点を当てています。

一方、Flexible Netflow レイヤ 3 機能は、ルーティングされたトラフィックのみをキャプチャし、スイッチングされたトラフィックはキャプチャしません。これは、Flexible NetFlow で最も一般的に使用されるアプリケーションの一つです。

ステートフル スイッチオーバー

Flexible NetFlow ステートフル スイッチオーバー (SSO) 機能により、NetFlow がトラフィックのモニタリングおよび分析に使用される環境で、高い可用性と信頼性が実現されます。この機能により、冗長化されたスーパーバイザ間でフェールオーバーが発生した場合でも、NetFlow データの収集およびエクスポートのプロセスがシームレスに続行されます。

この機能は、冗長化が有効になっており、冗長化コンフィギュレーション モードで **mode sso** コマンドを有効にすることによって動作モードが SSO に設定されている場合、デフォルトで有効になります。

Flexible NetFlow の前提条件

Flexible NetFlow の前提条件は、次のとおりです。

キーフィールド

次のコマンドで定義される Flexible NetFlow のキーフィールドについてよく理解していること。

- **match flow**
- **match interface**
- **match {ipv4 | ipv6}**
- **match routing**
- **match transport**

非キーフィールド

次のコマンドで定義される Flexible NetFlow の非キーフィールドについてよく理解していること。

- **collect counter**
- **collect timestamp absolute**
- **collect transport**（Cisco C9610 シリーズ スマートスイッチではサポートされていません）

IPv4 および IPv6 トラフィック

- ネットワーキングデバイスが、IPv4 および IPv6 ルーティング用に設定されていること。

Flexible Netflow に関する制約事項

共通の制約事項

- 出力トラフィックの場合、AVC または FNF のフロー学習は、パケットが FNF または AVC の学習パスと一致する場合にのみ発生します。出力 ACL の拒否ポリシーによって、学習またはミラーリングの前にパケットがドロップされる場合、FNF フローレコードは作成されず、そのフローに対する AVC または NBAR の分類は実行されません。
- システムは、最大 8 つの入力および 8 つの出力の NetFlow プロファイルをサポートします。
- キャッシュサイズや非アクティブタイムアウトなどのフローモニターのパラメータにより、NetFlow プロファイルが定義されます。
- 同じパラメータを使用するすべてのフローモニターが、同じ ASIC/スライスの組み合わせに属するポート上の入力または出力について NetFlow プロファイルを共有します。

- Flexible NetFlow は、レイヤ 2 およびレイヤ 3 のポートチャネル インターフェイスでサポートされていますが、メンバーポートではサポートされていません。
- Traditional NetFlow のアカウンティングはサポートされていません。
- 同じトラフィックタイプの複数のフローモニターを特定のインターフェイスおよび方向に適用することはできません。
- Flexible NetFlow エクスポートは、イーサネット管理ポート（GigabitEthernet 0/0）ではサポートされていません。
- 送信元グループタグ（SGT）フィールドと宛先グループタグ（DGT）フィールドはサポートされていません。
- NetFlow レコードは、マルチプロトコル ラベル スイッチング対応（MPLS 対応）インターフェイスをサポートしません。
- 出力 SVI およびサブインターフェイスに接続されたフローモニターでのインターフェイス名の照合により、SVI またはサブインターフェイス名ではなく、物理インターフェイス名が取得されます。
- レイヤ 3 インターフェイスの Flexible NetFlow は、ルーティングされたトラフィックのみを学習します。
- **match datalink vlan output** コマンドは、出力 Flexible NetFlow のデータリンクフローモニターではサポートされていません。
- ARP や CDP などの制御パケットは、Flexible NetFlow によって学習されません。
- フローモニターがインターフェイスに接続されている場合、アクティブタイムアウト、非アクティブタイムアウト、キャッシュサイズなどのフローモニターのパラメータは変更できません。
- パケット長に含まれる 4 バイトの巡回冗長検査は、バイト数の計算には含まれません。
- 出力側のルーティングされたトラフィックの場合、バイト数には IP ヘッダー長のみが含まれます。イーサネットヘッダー長は除外されます。
- IGMP タイプのキーフィールドもサポートされていません。
- レイヤ 3 インターフェイスでのデータリンクモニターはサポートされていません。
- MAC フィールドは、IPv4 および IPv6 フローモニターではサポートされていません。
- BGP ルーティングフィールドはサポートされていません。
- Flexible Netflow は、Cisco SD-Access（SDA）ではサポートされていません。
- サンプルリングは、バーストトラフィックにのみ適用されます。

Cisco C9350 シリーズ スマートスイッチに関する制約事項

- レイヤ 2 および VLAN の Flexible NetFlow はマルチキャストトラフィックを学習できますが、レイヤ 3 の Flexible NetFlow はマルチキャストトラフィックフローを学習しません。
- Flexible NetFlow とネットワーク アドレス変換（NAT）がインターフェイスで設定されている場合、

- Flexible NetFlow は、実際のフローの詳細を表示およびエクスポートします。変換されたフローの詳細は含まれません。アプリケーションレベル ゲートウェイ（ALG）フローの詳細は、エクスポートされる変換前のフローの詳細には含まれません。
- ALG トラフィックが CPU を介して変換されると、Flexible NetFlow は、ALG トラフィックの変換後のフローの詳細を表示およびエクスポートします。
- Network Based Application Recognition（NBAR）Flexible NetFlow レコードと通常の Flexible NetFlow レコードを同時に設定することはできません。
- 1つの ASIC で、入力および出力を含めて最大 8 つの NetFlow プロファイルがサポートされます。NetFlow プロファイルは、キャッシュサイズや非アクティブタイムアウトなどのフローモニターのパラメータによって定義されます。同じフローモニターパラメータを使用するすべてのフローモニターが、入力および出力について NetFlow プロファイルを共有します。たとえば、2 つのフローモニターが同じ NetFlow プロファイルを共有している場合、キャッシュサイズはこれら 2 つのフローモニター間で共有されます。
- 入力レイヤ 3 および SVI インターフェイスでのマルチキャストトラフィックの VRF ID はゼロです。

Cisco C9550 および C9610 シリーズ スマートスイッチに関する制約事項

- 入力プロファイルと出力プロファイルは個別のものであり、方向ごとに管理されます。これらはリソースを共有しません。
- バイトカウンタはサポートされていません。
- レイヤ 2 およびレイヤ 3 のマルチキャストトラフィックでの Flexible Netflow はサポートされていません。
- NAT はサポートされていません。
- NBAR Flexible Netflow はサポートされていません。
- システムでは、入力で最大 8 つの NetFlow プロファイルと出力で最大 8 つの NetFlow プロファイルがサポートされています。NetFlow プロファイルは、キャッシュサイズや非アクティブタイムアウトなどのフローモニターのパラメータによって定義されます。同じフローモニターパラメータを使用するすべてのフローモニターが、同じ ASIC/スライスの組み合わせに属するポートの入力または出力について NetFlow プロファイルを共有します。

Flexible NetFlow の設定

Flexible Netflow を設定するには、次の一般的な手順に従います。

1. フローにキー フィールドおよび非キー フィールドを指定して、フロー レコードを作成します。
2. プロトコルを指定して任意のフローエクスポートを作成し、宛先ポート、宛先、およびその他のパラメータを転送します。
3. フロー レコードおよびフロー エクスポートに基づいて、フロー モニターを作成します。
4. 任意のサンプラーを作成します。
5. レイヤ 2 ポート、レイヤ 3 ポート、または VLAN にフロー モニターを適用します。

さらに、その他の Flexible Netflow 機能を設定できます。

ユーザー定義フローレコードの作成および変更

ニーズに合わせてカスタマイズしたフローレコードを設定するには、このタスクを実行します。

カスタマイズしたフローレコードは、特定の目的でトラフィックデータを分析するために使用します。カスタマイズしたフローレコードには、key フィールドとして使用する **match** 基準が 1 つ以上必要です。通常は **nonkey** フィールドとして使用する **collect** 基準が 1 つ以上あります。

カスタマイズされたフローレコードは、特定のトラフィック分析ニーズを満たすためにさまざまな方法で設定できます。このタスクでは、可能性のある順列の 1 つを作成するための手順について説明します。要件に合わせてカスタマイズされたフローレコードを作成するために、これらの手順を変更できます。

	Command or Action	Purpose
Step 1	enable Example: 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）。
Step 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。
Step 3	flow record record-name Example: Device(config)# flow record FLOW-RECORD-1	フローレコードを作成し、Flexible NetFlow フローレコードコンフィギュレーションモードを開始します。 このコマンドでは、既存のフローレコードを変更することもできます。
Step 4	description description Example: Device(config-flow-record)# description Used for basic traffic analysis	（任意）フローレコードの説明を作成します。
Step 5	match {ipv4 ipv6} {destination source} address Example: Device(config-flow-record)# match ipv4 destination address	フローレコードの key フィールドを設定します。 Note この例では、IPv4宛先アドレスをレコードの key フィールドとして設定します。
Step 6	必要なすべてのキーフィールドがレコードに設定されるまで、必要に応じてステップ 5 を繰り返します。	—
Step 7	end Example: Device(config-flow-record)# end	Flexible NetFlow フローレコード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。

	Command or Action	Purpose
Step 8	show flow record <i>record-name</i> Example: Device# show flow record FLOW_RECORD-1	(任意) 指定したフロー レコードの現在のステータスが表示されます。
Step 9	show running-config flow record <i>record-name</i> Example: Device# show running-config flow record FLOW_RECORD-1	(任意) 指定したフロー レコードの設定が表示されます。

フローエクスポートの作成

フロー エクスポートを作成して、フローのエクスポート パラメータを定義できます。

Note

フロー エクスポートごとに、1 つの宛先のみがサポートされます。複数の宛先にデータをエクスポートする場合は、複数のフロー エクスポートを設定してフロー モニターに割り当てる必要があります。

IPv4 アドレスを使用して宛先にエクスポートできます。

	Command or Action	Purpose
Step 1	enable Example: 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)。
Step 2	configure terminal Example: Device(config)# 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。
Step 3	flow exporter name Example: Device(config)# flow exporter ExportTest	フロー エクスポートを作成し、フロー エクスポート コンフィギュレーション モードを開始します。
Step 4	description string Example: Device(config-flow-exporter)# description ExportV9	(任意) 最大 63 文字で、このフローの説明を指定します。
Step 5	destination {ipv4-address} Example: Device(config-flow-exporter)# destination 192.0.2.1 (IPv4 destination)	このエクスポートに IPv4 宛先アドレスまたはホスト名を設定します。
Step 6	dscp value Example:	(任意) DSCP (DiffServ コードポイント) 値を指定します。範囲は 0～63 です。デフォルトは 0 です。

	Command or Action	Purpose
	Device(config-flow-exporter)# dscp 0	
Step 7	source <i>interface type interface number</i> Example: Device(config-flow-exporter)# source gigabitEthernet1/0/1	(任意) 設定された宛先で NetFlow コネクタに到達するために使用するインターフェイスを指定します。  Note フローエクスポートは、送信元インターフェイスとしてアンナンバード IP インターフェイスをサポートしていません。 送信元として次のインターフェイスを設定できます。 • Auto Template: 自動テンプレートインターフェイス • Capwap: Capwap トンネルインターフェイス • GigabitEthernet: Gigabit Ethernet IEEE 802 • GroupVI: グループ仮想インターフェイス • Internal Interface: 内部インターフェイス • Loopback: ループバック インターフェイス • Null: ノルインターフェイス • Port-channel: インターフェイスのイーサネットチャネル • TenGigabitEthernet: 10 ギガビットイーサネット • Tunnel: トンネルインターフェイス • Vlan: Catalyst VLAN
Step 8	transport udp <i>number</i> Example: Device(config-flow-exporter)# transport udp 200	(任意) NetFlow コレクタに到達するために使用する UDP ポートを指定します。
Step 9	ttl <i>seconds</i> Example: Device(config-flow-exporter)# ttl 210	(任意) エクスポートによって送信されるデータグラムの存続可能時間 (TTL) 値を設定します。範囲は 1 ~ 255 秒です。デフォルトは 255 です。
Step 10	export-protocol {netflow-v9} Example:	エクスポートで使用する NetFlow エクスポートプロトコルのバージョンを指定します。

	Command or Action	Purpose
	Device(config-flow-exporter)# export-protocol netflow-v9	
Step 11	end Example: Device(config-flow-record)# end	特権 EXEC モードに戻ります。
Step 12	show flow exporter [name record-name] Example: Device# show flow exporter ExportTest	(任意) NetFlow のフロー エクスポート情報を表示します。
Step 13	copy running-config startup-config Example: Device# copy running-config startup-config	(任意) コンフィギュレーションファイルに設定を保存します。

カスタマイズしたフローモニターを作成します

各フロー モニターには、専用のキャッシュが割り当てられています。フローモニターのパラメータ（キャッシュサイズや非アクティブタイムアウトなど）は、NetFlow プロファイルと呼ばれます。フロー モニタごとに、キャッシュエントリの内容およびレイアウトを定義するレコードが必要です。上級のユーザーであれば、**flow record** コマンドを使用して、カスタマイズしたフォーマットを作成することもできます。

- カスタマイズしたレコードを使用する場合は、このタスクを実行する前に、カスタマイズしたレコードを作成する必要があります。
- データをエクスポートするためにフロー エクスポートをフロー モニターに追加する場合は、このタスクを完了する前にエクスポートを作成する必要があります。

Note

フローモニターで **record** コマンドのパラメータを変更する前に、**no ip flow monitor** コマンドを使用して、すべてのインターフェイスから適用済みのフローモニターを削除する必要があります。

	Command or Action	Purpose
Step 1	enable Example: 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）。
Step 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
Step 3	flow monitor <i>monitor-name</i> Example: Device(config)# flow monitor FLOW-MONITOR-1	フロー モニターを作成し、Flexible NetFlow フロー モニター コンフィギュレーション モードを開始します。 このコマンドでは、既存のフロー モニターを変更することもできます。
Step 4	description <i>description</i> Example: Device(config-flow-monitor)# description Used for basic ipv4 traffic analysis	(任意) フロー モニターの説明を作成します。
Step 5	record { <i>record-name</i> netflow-original netflow { ipv4 ipv6 } <i>record</i> [peer]} Example: Device(config-flow-monitor)# record FLOW-RECORD-1	フロー モニターのレコードを指定します。
Step 6	cache { timeout { active inactive update rate-limit } <i>seconds</i> type normal } Example: Device(config-flow-monitor)# cache type normal Device(config-flow-monitor)# cache timeout active	(任意) フローモニター キャッシュ パラメータ (タイムアウト値、キャッシュタイプなど) を変更します。指定したフロー モニターとフロー キャッシュを関連付けます。
Step 7	必要に応じて、ステップ 6 を繰り返して、このフロー モニターのキャッシュ パラメータの変更を完了します。	—
Step 8	exporter <i>exporter-name</i> Example: Device(config-flow-monitor)# exporter EXPORTER-1	(任意) 事前に作成されたエクスポートの名前を指定します。
Step 9	end Example: Device(config-flow-monitor)# end	Flexible NetFlow フロー モニター 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。
Step 10	show flow monitor [[name] <i>monitor-name</i> [cache [format { csv record table }]] [statistics]] Example: Device# show flow monitor FLOW-MONITOR-2 cache	(任意) Flexible NetFlow フロー モニターのステータスおよび統計情報が表示されます。
Step 11	show running-config flow monitor <i>monitor-name</i> Example: Device# show running-config flow monitor FLOW_MONITOR-1	(任意) 指定したフロー モニターの設定が表示されます。

	Command or Action	Purpose
Step 12	copy running-config startup-config Example: Device# copy running-config startup-config	(任意) コンフィギュレーションファイルに設定を保存します。

フローサンプラの作成

フロー サンプラーを設定して有効化するには、この必須のタスクを実行します。

	Command or Action	Purpose
Step 1	enable Example: 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）。
Step 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。
Step 3	sampler sampler-name Example: Device(config)# sampler SAMPLER-1	サンプラーを作成し、サンプラー コンフィギュレーション モードを開始します。 このコマンドでは、既存のサンプラーを変更することもできます。
Step 4	description description Example: Device(config-sampler)# description Sample at 50%	(任意) フロー サンプラーの説明を作成します。
Step 5	mode {random} 1 out-of window-size Example: Device(config-sampler)# mode random 1 out-of 2	サンプラー モードおよびフロー サンプラーのウィンドウ サイズを指定します。 window-size 引数の範囲は、2 ～ 32576 です。
Step 6	exit Example: Device(config-sampler)# exit	サンプラー コンフィギュレーション モードを終了し、グローバル コンフィギュレーション モードに戻ります。
Step 7	interface type number Example: Device(config)# interface GigabitEthernet 1/0/1	インターフェイスを指定し、インターフェイス コンフィギュレーション モードを開始します。
Step 8	{ip ipv6} flow monitor monitor-name [[sampler] sampler-name] {input output} Example:	作成したフローモニターとサンプラをインターフェイスに割り当てて、サンプリングを有効にします。

	Command or Action	Purpose
	Device(config-if)# ip flow monitor FLOW-MONITOR-1 sampler SAMPLER-1 input	
Step 9	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。
Step 10	show sampler sampler-name Example: Device# show sampler SAMPLER-1	設定し有効化したフローサンプラーのステータスおよび統計情報を表示します。

インターフェイスへのフローの適用

フロー モニターおよびオプションのサンプラーをインターフェイスに適用できます。

	Command or Action	Purpose
Step 1	enable Example: 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）。
Step 2	configure terminal Example: Device(config)# 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。
Step 3	interface type Example: Device(config)# interface GigabitEthernet1/0/1	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、インターフェイスを設定します。
Step 4	{ip flow monitor ipv6 flow monitor datalink flow monitor} name [sampler name] {input output} Example: Device(config-if)# ip flow monitor MonitorTest input	入力または出力パケットに対応するインターフェイスに、IPv4、IPv6、およびデータリンクフローモニターと、オプションのサンプラーを関連付けます。 • ip flow monitor: Flexible NetFlow で IPv4 トラフィックを監視できます。 • ipv6 flow monitor: Flexible NetFlow で IPv6 トラフィックを監視できます。 • datalink flow monitor: Flexible NetFlow で非 IP のトラフィックを監視できます。  Note 入力と出力の両方向でインターフェイスに複数のモニターを関連付けることができます。

	Command or Action	Purpose
Step 5	end Example: Device(config-flow-monitor) # end	特権 EXEC モードに戻ります。
Step 6	show flow interface [<i>interface-type number</i>] Example: Device# show flow interface	(任意) インターフェイスの NetFlow 情報を表示します。
Step 7	copy running-config startup-config Example: Device# copy running-config startup-config	(任意) コンフィギュレーション ファイルに設定を保存します。

VLAN 上でのブリッジ型 NetFlow の設定

フローモニターおよびオプションのサンプラを VLAN に適用します。

	Command or Action	Purpose
Step 1	enable Example: 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)。
Step 2	configure terminal Example: Device(config) # 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。
Step 3	vlan [<i>configuration</i>] <i>vlan-id</i> Example: Device(config) # vlan configuration 30	VLAN を設定し、VLAN または VLAN コンフィギュレーション モードを開始します。
Step 4	ip flow monitor <i>monitor name</i> [sampler <i>sampler name</i>] {input} Example: Device(config-vlan-config) # ip flow monitor MonitorTest input	入力パケットに対応する VLAN に、フローモニターおよびオプションのサンプラを関連付けます。
Step 5	copy running-config startup-config Example: Device# copy running-config startup-config	(任意) コンフィギュレーション ファイルに設定を保存します。

Flexible NetFlow の入力および出力 VRF の設定

入力または出力フローモニターを適用することにより、デバイス上の着信パケットから VRF ID を収集するように設定するには、このタスクを実行します。

 Note

Cisco C9610 シリーズ スマートスイッチでは、VRF match フィールドは入力方向でのみサポートされています。

	Command or Action	Purpose
Step 1	enable Example: 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）。
Step 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。
Step 3	flow record record-name Example: Device(config)# flow record rm_1	フロー レコードを作成し、Flexible NetFlow フロー レコード コンフィギュレーション モードを開始します。 このコマンドでは、既存のフロー レコードを変更することもできます。
Step 4	match routing vrf input Example: Device(config-flow-record)# match routing vrf input	着信の Virtual Routing and Forwarding (VRF) ID をキーフィールドとして設定します。
Step 5	match {ip ipv6} {destination source} address Example: Device(config-flow-record)# match ipv4 destination address	フロー レコードの key フィールドを設定します。  Note この例では、IPv4 宛先アドレスをレコードの key フィールドとして設定します。
Step 6	必要なすべてのキーフィールドがレコードに設定されるまで、必要に応じてステップ 5 を繰り返します。	—
Step 7	exit Example: Device(config-flow-record)# end	Flexible NetFlow フロー レコード コンフィギュレーション モードを終了して、グローバル コンフィギュレーション モードに戻ります。
Step 8	flow monitor monitor-name Example: Device(config)# flow monitor mm_1	フロー モニターを作成し、Flexible NetFlow フロー モニター コンフィギュレーション モードを開始します。 このコマンドでは、既存のフロー モニターを変更することもできます。

	Command or Action	Purpose
Step 9	record { <i>record-name</i> netflow-original netflow { ipv4 ipv6 } record [peer]} Example: Device(config-flow-monitor)# record rm_1	フロー モニターのレコードを指定します。
Step 10	exit Example: Device(config-flow-record)# end	Flexible NetFlow フロー レコード コンフィギュレーションモードを終了して、グローバルコンフィギュレーション モードに戻ります。
Step 11	interface GigabitEthernet <i>interface-id</i> Example: Device(config)# interface GigabitEthernet 1/0/1	インターフェイスを作成して、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。
Step 12	ip vrf forwarding <i>vrf-name</i> Example: Device(config-if)# ip vrf forwarding green	VRF インスタンスをインターフェイスに関連付けます。
Step 13	ip address <i>ip-address</i> Example: Device(config-if)# ip address 172.16.2.2 255.255.255.252	インターフェイスの IP アドレスを設定します。
Step 14	ip flow monitor <i>monitor-name</i> { input output } Example: Device(config-if)# ip flow monitor mm_1 input	IPv4 トラフィックの Flexible NetFlow フローモニターを有効にします。
	 Note	次の例では、デバイスが受信する IPv4 トラフィックのフローモニターが有効になります。
Step 15	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。

レイヤ 2 NetFlow の設定

Flexible NetFlow レコード内でレイヤ 2 キーを定義して、レイヤ 2 インターフェイスのフローをキャプチャできます。

	Command or Action	Purpose
Step 1	enable Example: 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）。

	Command or Action	Purpose
Step 2	configure terminal Example: Device(config)# 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。
Step 3	flow record name Example: Device(config)# flow record L2_record	フローレコードコンフィギュレーションモードを開始します。
Step 4	match datalink {dot1q ethertype mac vlan} Example: Device(config-flow-record)# match datalink ethertype	レイヤ 2 属性をキーとして指定します。
Step 5	end Example: Device(config-flow-record)# end	特権 EXEC モードに戻ります。
Step 6	show flow record [name] Example: Device# show flow record	(任意) インターフェイスの NetFlow 情報を表示します。
Step 7	copy running-config startup-config Example: Device# copy running-config startup-config	(任意) コンフィギュレーション ファイルに設定を保存します。

レイヤ 3 NetFlow の設定

Flexible NetFlow レコード内でレイヤ 3 キーを定義して、レイヤ 3 インターフェイスのフローをキャプチャできます。

	Command or Action	Purpose
Step 1	enable Example: 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)。
Step 2	configure terminal Example: Device(config)# 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。
Step 3	flow record name Example: Device(config)# flow record L3_record	フローレコードコンフィギュレーションモードを開始します。

	Command or Action	Purpose
Step 4	match ipv4 destination address Example: Device(config-flow-record) # match ipv4 destination address	レイヤ 3 属性をキーとして指定します。
Step 5	end Example: Device(config-flow-record) # end	特権 EXEC モードに戻ります。
Step 6	show flow record [name] Example: Device# show flow record	(任意) インターフェイスの NetFlow 情報を表示します。
Step 7	copy running-config startup-config Example: Device# copy running-config startup-config	(任意) コンフィギュレーション ファイルに設定を保存します。

設定例

Flexible NetFlow の設定例については、このセクションを参照してください。

例：フローの設定とインターフェイスへの適用

フローを作成し、そのフローをインターフェイスに適用する例を示します。

```
Device# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.

Device(config)# flow export export1
Device(config-flow-exporter)# destination 10.0.101.254
Device(config-flow-exporter)# transport udp 2055
Device(config-flow-exporter)# exit
Device(config)# flow record record1
Device(config-flow-record)# match ipv4 source address
Device(config-flow-record)# match ipv4 destination address
Device(config-flow-record)# match ipv4 protocol
Device(config-flow-record)# match transport source-port
Device(config-flow-record)# match transport destination-port
Device(config-flow-record)# collect counter
Device(config-flow-record)# collect timestamp absolute first
Device(config-flow-record)# collect timestamp absolute last
Device(config-flow-record)# exit
Device(config)# flow monitor monitor1
Device(config-flow-monitor)# record record1
Device(config-flow-monitor)# exporter export1
Device(config-flow-monitor)# exit
Device(config)# interface tenGigabitEthernet 1/0/1
Device(config-if)# ip flow monitor monitor1 input
Device(config-if)# end
```

例：VLAN 上でのブリッジ型 NetFlow の設定

次の例は、VLAN でブリッジ型 NetFlow を設定する方法を示しています。

```
Device# configure terminal
Device(config)# vlan configuration 30
Device(config-vlan-config)# ip flow monitor MonitorTest input
Device(config-vlan-config)# end
```

例：入力 VRF 用の Flexible NetFlow の設定

次の例では、VRF ID をキーフィールドとして収集するフローレコードを持つ入力フローモニターを適用することで、デバイスの着信パケットからの VRF ID の収集を設定します。

```
Device> enable
Device# configure terminal
Device(config)# flow record rm_1
Device(config-flow-record)# match routing vrf input
Device(config-flow-record)# match ipv4 source address
Device(config-flow-record)# match ipv4 destination address
Device(config-flow-record)# collect counter
Device(config-flow-record)# exit

Device(config)# flow monitor mm_1
Device(config-flow-record)# record rm_1
Device(config-flow-record)# exit

Device(config)# interface GigabitEthernet 1/0/1
Device(config-if)# ip vrf forwarding green
Device(config-if)# ip address 172.16.2.2 255.255.255.252
Device(config-if)# ip flow monitor mm_1 input
Device(config-if)# end
```

例：出力 VRF 用の Flexible NetFlow の設定

次の例では、VRF ID をキーフィールドとして収集するフローレコードを持つ出力フローモニターを適用することで、デバイスの発信パケットからの VRF ID の収集を設定します。

Note

次の例は、Cisco C9610 シリーズ スマートスイッチには適用されません。

```
Device> enable
Device# configure terminal
Device(config)# flow record rm_1
Device(config-flow-record)# match routing vrf input
Device(config-flow-record)# match ipv4 source address
Device(config-flow-record)# match ipv4 destination address
Device(config-flow-record)# collect counter
Device(config-flow-record)# exit

Device(config)# flow monitor mm_1
Device(config-flow-record)# record rm_1
Device(config-flow-record)# exit

Device(config)# interface GigabitEthernet 1/0/1
Device(config-if)# ip vrf forwarding green
Device(config-if)# ip address 172.16.2.2 255.255.255.252
```

```
Device(config-if)# ip flow monitor mm_1 output
Device(config-if)# end
```

例：レイヤ 2 NetFlow の設定

この例では、レイヤ 2 NetFlow の設定方法を示します。

```
Device# configure terminal
Device(config)# flow record L2_record
Device(config-flow-record)# match datalink ethertype
Device(config-flow-record)# end
```

例：レイヤ 3 NetFlow の設定

この例では、レイヤ 3 NetFlow の設定方法を示します。

```
Device# configure terminal
Device(config)# flow record L3_record
Device(config-flow-record)# match ipv4 destination address
Device(config-flow-record)# end
```

Flexible NetFlow の監視

次の表にあるコマンドを使用して、Flexible NetFlow をモニタリングできます。

Table 4: Flexible NetFlow の監視コマンド

コマンド	目的
show redundancy [clients counters history switchover history states]	SSO 設定に関する情報を表示します。
show flow exporter [broker export-ids name name statistics templates]	NetFlow のフローエクスポートと統計に関する情報を表示します。
show flow interface	NetFlow インターフェイスに関する情報を表示します。
show flow monitor [name exporter-name]	NetFlow のフローモニターと統計に関する情報を表示します。
show flow monitor name statistics	フローモニターの統計を表示します。
show flow monitor name cache format {table record csv}	フローモニターのキャッシュの内容を指定された形式で表示します。
show flow record [name record-name]	NetFlow のフローレコードに関する情報を表示します。
show sampler [broker name name]	NetFlow サンプラに関する情報を表示します。
show platform software fed switch active sdm feature	ASIC ごと、スライスごと、および方向ごとの NetFlow フローに関する情報を表示します。

例: IPv4 入力トラフィックの監視

次の例は、IPv4 入力トラフィックをモニターする方法を示しています（int g1/0/11 は、int g1/0/36 および int g3/0/11 にトラフィックを送信します）。

```
Device# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Device(config)# flow record fr-1
Device(config-flow-record)# match ipv4 source address
Device(config-flow-record)# match ipv4 destination address
Device(config-flow-record)# match interface input
Device(config-flow-record)# collect counter
Device(config-flow-record)# collect timestamp absolute first
Device(config-flow-record)# collect timestamp absolute last
Device(config-flow-record)# exit

Device(config)# flow exporter fe-ipfix6
Device(config-flow-exporter)# destination 2001:0:0:24::10
Device(config-flow-exporter)# source Vlan106
Device(config-flow-exporter)# transport udp 4739
Device(config-flow-exporter)# export-protocol ipfix
Device(config-flow-exporter)# template data timeout 240
Device(config-flow-exporter)# exit

Device(config)# flow exporter fe-ipfix
Device(config-flow-exporter)# description IPFIX format collector 100.0.0.80
Device(config-flow-exporter)# destination 100.0.0.80
Device(config-flow-exporter)# dscp 30
Device(config-flow-exporter)# ttl 210
Device(config-flow-exporter)# transport udp 4739
Device(config-flow-exporter)# export-protocol ipfix
Device(config-flow-exporter)# template data timeout 240
Device(config-flow-exporter)# exit

Device(config)# flow exporter fe-1
Device(config-flow-exporter)# destination 10.5.120.16
Device(config-flow-exporter)# source Vlan105
Device(config-flow-exporter)# dscp 32
Device(config-flow-exporter)# ttl 200
Device(config-flow-exporter)# transport udp 2055

Device(config-flow-exporter)# template data timeout 240
Device(config-flow-exporter)# exit

Device(config)# flow monitor fm-1
Device(config-flow-monitor)# exporter fe-ipfix6
Device(config-flow-monitor)# exporter fe-ipfix
Device(config-flow-monitor)# exporter fe-1
Device(config-flow-monitor)# cache timeout inactive 60
Device(config-flow-monitor)# cache timeout active 180
Device(config-flow-monitor)# record fr-1
Device(config-flow-monitor)# end

Device# show running-config interface g1/0/11
Device# show running-config interface g1/0/36
Device# show running-config interface g3/0/11
Device# show flow monitor fm-1 cache format table
```

例: IPv4 出力トラフィックの監視

次の例は、IPv4 出力トラフィックをモニターする方法を示しています。

```
Device# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Device(config)# flow record fr-1 out
Device(config-flow-record)# match ipv4 source address
Device(config-flow-record)# match ipv4 destination address
Device(config-flow-record)# match interface output
Device(config-flow-record)# collect counter
Device(config-flow-record)# collect timestamp absolute first
Device(config-flow-record)# collect timestamp absolute last
Device(config-flow-record)# exit

Device(config)# flow exporter fe-1
Device(config-flow-exporter)# destination 10.5.120.16
Device(config-flow-exporter)# source Vlan105
Device(config-flow-exporter)# dscp 32
Device(config-flow-exporter)# ttl 200
Device(config-flow-exporter)# transport udp 2055
Device(config-flow-exporter)# template data timeout 240
Device(config-flow-exporter)# exit

Device(config)# flow exporter fe-ipfix6
Device(config-flow-exporter)# destination 2001:0:0:24::10
Device(config-flow-exporter)# source Vlan106
Device(config-flow-exporter)# transport udp 4739
Device(config-flow-exporter)# export-protocol ipfix
Device(config-flow-exporter)# template data timeout 240
Device(config-flow-exporter)# exit

Device(config)# flow exporter fe-ipfix
Device(config-flow-exporter)# description IPFIX format collector 100.0.0.80
Device(config-flow-exporter)# destination 100.0.0.80
Device(config-flow-exporter)# dscp 30
Device(config-flow-exporter)# ttl 210
Device(config-flow-exporter)# transport udp 4739
Device(config-flow-exporter)# export-protocol ipfix
Device(config-flow-exporter)# template data timeout 240
Device(config-flow-exporter)# exit

Device(config)# flow monitor fm-1-output
Device(config-flow-monitor)# exporter fe-1
Device(config-flow-monitor)# exporter fe-ipfix6
Device(config-flow-monitor)# exporter fe-ipfix
Device(config-flow-monitor)# cache timeout inactive 50
Device(config-flow-monitor)# cache timeout active 120
Device(config-flow-monitor)# record fr-1-out
Device(config-flow-monitor)# end

Device# show flow monitor fm-1-output cache format table
```

翻訳について

このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。