



## NetFlow の設定

この章では、NetFlow を設定して、送信元、トラフィック宛先、タイミング、アプリケーション情報に基づいて、IP トラフィックを特徴付けます。これによって、仮想スイッチを通過するトラフィックを可視化します。この情報を使用して、ネットワーク可用性とパフォーマンスを評価し、規制要件（コンプライアンス）を満たすことができます。また、トラブルシューティングを支援することもできます。

ここでは、次の内容について説明します。

- 「NetFlow 情報」 (P.11-1)
- 「NetFlow の前提条件」 (P.11-9)
- 「設定時の注意事項および制約事項」 (P.11-9)
- 「NetFlow の設定」 (P.11-10)
- 「NetFlow の設定確認」 (P.11-21)
- 「NetFlow の設定例」 (P.11-19)
- 「デフォルト設定」 (P.11-24)
- 「その他の関連資料」 (P.11-25)

## NetFlow 情報

NetFlow は IP トラフィックを評価し、IP トラフィックがフローする方法と場所を把握できます。NetFlow は、アカウントティング、ネットワーク モニタリング、およびネットワーク プランニングで使用できる情報を収集します。

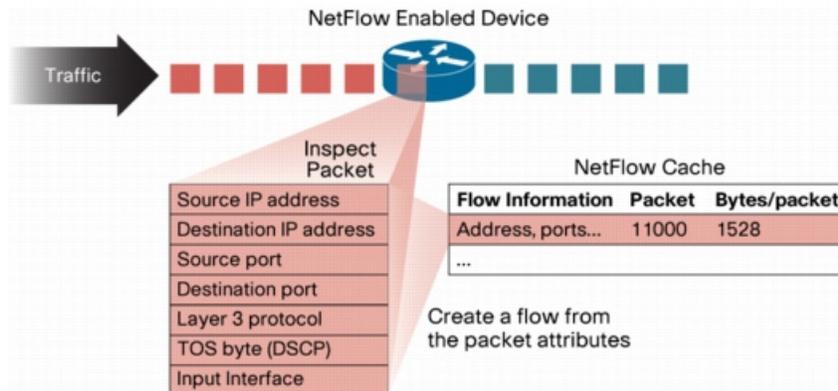
ここでは、次の内容について説明します。

- 「フローとは何か」 (P.11-2)
- 「フロー レコード定義」 (P.11-2)
- 「NetFlow で生成されるフロー レコードにアクセスする方法」 (P.11-5)
- 「NetFlow コレクタ サーバへのフローのエクスポート」 (P.11-7)
- 「NetFlow データの例」 (P.11-8)
- 「ハイ アベイラビリティ」 (P.11-9)

## フローとは何か

フローとは、送信元インターフェイス（またはサブインターフェイス）に到着する一方向のパケットストリームであり、一連の条件と一致します。同じ送信元/宛先 IP アドレス、送信元/宛先ポート、プロトコル インターフェイス、サービス クラスを持つすべてのパケットは、フローにグループ化され、パケットとバイト数が計算されます。これにより、大量のネットワーク情報が、NetFlow キャッシュと呼ばれるデータベースに集約されます。

図 11-1 NetFlow キャッシュでのフローの作成



NetFlow が収集する条件を定義して、フローを作成できます。フローは NetFlow キャッシュに格納されます。

フロー情報には次の内容が含まれます。

- 送信元アドレスは、トラフィックを送信したユーザを示します。
- 宛先アドレスは、トラフィックを受信するユーザを示します。
- ポートはトラフィックを使用するアプリケーションを特定します。
- サービス クラスは、トラフィックのプライオリティを調べます。
- デバイス インターフェイスは、トラフィックがネットワーク デバイスで使用される方法を示します。
- 計算されたパケットとバイト数は、トラフィック量を示します。

## フロー レコード定義

フロー レコードは、フロー内のパケットやフローごとに収集されたカウンタのタイプなど、NetFlow が収集する情報を定義します。新しいフロー レコードを定義するか、あらかじめ定義された Cisco Nexus 1000V フロー レコードを使用できます。

レコードを作成するには、「[フロー レコードの定義](#)」の手順 (P.11-10) を参照してください。

次の表では、フロー レコードで定義される条件を説明します。

フロー レコード条件	説明
Match	<p>フロー レコードのコレクションと一致する情報を定義します。</p> <ul style="list-style-type: none"> <li>• <b>ip</b> : フロー レコードで収集されるデータは、次の IP オプションのいずれかと一致します。 <ul style="list-style-type: none"> <li>- protocol</li> <li>- tos (タイプ オブ サービス)</li> </ul> </li> <li>• <b>ipv4</b> : フロー レコードで収集されるデータは、次の ipv4 アドレス オプションのいずれかと一致します。 <ul style="list-style-type: none"> <li>- 送信元アドレス</li> <li>- 宛先アドレス</li> </ul> </li> <li>• <b>transport</b> : フロー レコードで収集されるデータは、次のトランスポート オプションのいずれかと一致します。 <ul style="list-style-type: none"> <li>- 宛先ポート</li> <li>- 送信元ポート</li> </ul> </li> </ul>
Collect	<p>フロー レコードが情報を収集する方法を定義します。</p> <ul style="list-style-type: none"> <li>• <b>counter</b> : 次のフォーマットのいずれかで、フロー レコード情報を収集します。 <ul style="list-style-type: none"> <li>- <b>bytes</b> : 32 ビットのカウンタで収集されます (ロングの 64 ビット カウンタが指定されている場合を除く)。</li> <li>- <b>packets</b> : 32 ビットのカウンタで収集されます (ロングの 64 ビット カウンタが指定されている場合を除く)。</li> </ul> </li> <li>• <b>timestamp sys-uptime</b> : フローの先頭または最終パケットに関するシステム稼動時間を収集します。</li> <li>• <b>transport tcp flags</b> : フローのパケットの TCP トランスポート レイヤフラグを収集します。</li> </ul>

## あらかじめ定義されたフロー レコード

Cisco Nexus 1000V には、次のあらかじめ定義されたフロー レコードが含まれます。

- 「Cisco Nexus 1000V であらかじめ定義されたフロー レコード : Netflow-Original」 (P.11-3)
- 「Cisco Nexus 1000V であらかじめ定義されたフロー レコード : Netflow IPv4 Original-Input」 (P.11-4)
- 「Cisco Nexus 1000V であらかじめ定義されたフロー レコード : Netflow IPv4 Original-Output」 (P.11-5)
- 「Cisco Nexus 1000V であらかじめ定義されたフロー レコード : Netflow IPv4 Protocol-Port」 (P.11-5)

### 例 11-1 Cisco Nexus 1000V であらかじめ定義されたフロー レコード : Netflow-Original

```
n1000v# show flow record netflow-original
Flow record netflow-original:
  Description: Traditional IPv4 input NetFlow with origin ASs
```

```

No. of users: 0
Template ID: 0
Fields:
  match ipv4 source address
  match ipv4 destination address
  match ip protocol
  match ip tos
  match transport source-port
  match transport destination-port
  match interface input
  match interface output
  match flow direction
  collect routing source as
  collect routing destination as
  collect routing next-hop address ipv4
  collect transport tcp flags
  collect counter bytes
  collect counter packets
  collect timestamp sys-uptime first
  collect timestamp sys-uptime last
n1000v#

```



**(注)** **show flow record** コマンドによる出力結果には、次の行が表示されますが、基になるコマンドは現在 Cisco Nexus 1000V ではサポートされていません。これらのコマンドを使用しても、コンフィギュレーションには影響はありません。

```

collect routing source as
collect routing destination as
collect routing next-hop address ipv4

```

#### 例 11-2 Cisco Nexus 1000V であらかじめ定義されたフロー レコード : Netflow IPv4 Original-Input

```

n1000v# show flow record netflow ipv4 original-input
Flow record ipv4 original-input:
  Description: Traditional IPv4 input NetFlow
  No. of users: 0
  Template ID: 0
  Fields:
    match ipv4 source address
    match ipv4 destination address
    match ip protocol
    match ip tos
    match transport source-port
    match transport destination-port
    match interface input
    match interface output
    match flow direction
    collect routing source as
    collect routing destination as
    collect routing next-hop address ipv4
    collect transport tcp flags
    collect counter bytes
    collect counter packets
    collect timestamp sys-uptime first
    collect timestamp sys-uptime last
n1000v#

```

**例 11-3 Cisco Nexus 1000V であらかじめ定義されたフロー レコード : Netflow IPv4 Original-Output**

```
switch# show flow record netflow ipv4 original-output
Flow record ipv4 original-output:
  Description: Traditional IPv4 output NetFlow
  No. of users: 0
  Template ID: 0
  Fields:
    match ipv4 source address
    match ipv4 destination address
    match ip protocol
    match ip tos
    match transport source-port
    match transport destination-port
    match interface input
    match interface output
    match flow direction
    collect routing source as
    collect routing destination as
    collect routing next-hop address ipv4
    collect transport tcp flags
    collect counter bytes
    collect counter packets
    collect timestamp sys-uptime first
    collect timestamp sys-uptime last
switch#
```

**例 11-4 Cisco Nexus 1000V であらかじめ定義されたフロー レコード : Netflow IPv4 Protocol-Port**

```
switch# show flow record netflow ipv4 protocol-port
Flow record ipv4 protocol-port:
  Description: Protocol and Ports aggregation scheme
  No. of users: 0
  Template ID: 0
  Fields:
    match ip protocol
    match transport source-port
    match transport destination-port
    match interface input
    match interface output
    match flow direction
    collect counter bytes
    collect counter packets
    collect timestamp sys-uptime first
    collect timestamp sys-uptime last
switch#
```

## NetFlow で生成されるフロー レコードにアクセスする方法

NetFlow データには、主に 2 つの方法を使用してアクセスできます。

- 「[コマンドライン インターフェイス \(CLI\)](#)」 (P.11-6)
- 「[NetFlow コレクタ](#)」 (P.11-7)

## コマンドライン インターフェイス (CLI)

現在のネットワーク状態を表示するには、CLI を使用できます。NetFlow CLI はトラブルシューティングの際に非常に有用です。利用可能な show コマンドを表示するには、「[NetFlow の設定確認 \(P.11-21\)](#)」を参照してください。

CLI は次のツールを使用して、フロー レコードを取得して、Netflow コレクタにエクスポートします。

- 「[フロー モニタ \(P.11-6\)](#)」
- 「[フロー エクスポート \(P.11-6\)](#)」

## フロー モニタ

フロー モニタは、次の NetFlow コンポーネント間のアソシエーションを作成します。

- フロー レコード：マッチングおよび収集条件から構成されます。
- フロー エクスポート：エクスポート条件から構成されます。

このフロー モニタ アソシエーションでは、レコードおよびエクスポートから構成されるセットを定義し、それを何度も再利用できます。さまざまなニーズに合わせて、複数のフロー モニタを作成できます。フロー モニタは、特定の方向の、特定のインターフェイスに適用されます。

「[フロー モニタの定義](#)」の手順 (P.11-16) および「[フロー モニタのインターフェイスへの割り当て](#)」の手順 (P.11-18) を参照してください。

## フロー エクスポート

フロー エクスポートを使用して、キャッシュから NetFlow コレクタと呼ばれるレポートिंग サーバに、フロー レコードが送信される場所と時間を定義します。

エクスポート定義には、次の内容が含まれます。

- 宛先 IP アドレス
- 送信元インターフェイス
- UDP ポート番号 (コレクタが待機するところ)
- エクスポート フォーマット



(注)

NetFlow エクスポート パケットでは、送信元インターフェイスに割り当てられた IP アドレスを使用します。送信元インターフェイスに IP アドレスが割り当てられていない場合、エクスポートはアクティブになりません。

「[フロー エクスポートの定義](#)」の手順 (P.11-13) を参照してください。

## エクスポート フォーマット

Cisco Nexus 1000V は、NetFlow バージョン 9 エクスポート フォーマットをサポートしています。



(注)

Cisco Nexus 1000V は、モニタごとに最大 2 つのエクスポートにデータをエクスポートするトランスポート プロトコルとして、UDP をサポートしています。

## NetFlow コレクタ

Cisco Nexus 1000V NetFlow キャッシュから、NetFlow コレクタと呼ばれるレポーティング サーバに、NetFlow をエクスポートできます。NetFlow コレクタは、エクスポートされたフローを再構成し、フローを組み合わせて、トラフィックおよびセキュリティ分析で使用するレポートを生成します。SNMP ポーリングとは異なり、NetFlow エクスポートは定期的に、情報を NetFlow レポーティング コレクタにプッシュします。NetFlow キャッシュには常にフローが入ってきます。Cisco Nexus 1000V はキャッシュ内の終了または期限切れになったフローを検索し、NetFlow コレクタ サーバにエクスポートします。ネットワーク通信が終了すると、フローも終了します。つまり、パケットには、TCP FIN フラグが含まれます。

次は、NetFlow データ レポーティングを実装する手順です。

- NetFlow が収集する情報を定義するように、NetFlow レコードを設定します。
- NetFlow キャッシュへのフロー レコードを取得するように、Netflow モニタを設定します。
- フローをコレクタに送信するように、NetFlow エクスポートを設定します。
- Cisco Nexus 1000V は、NetFlow キャッシュ内の終了したフローを検索し、NetFlow コレクタ サーバにエクスポートします。
- フローは UDP エクスポート パケットで利用可能なスペース、またはエクスポート タイマーに基づいてまとめられます。
- NetFlow コレクタ ソフトウェアは、データからリアルタイムのレポートまたは履歴レポートを作成します。

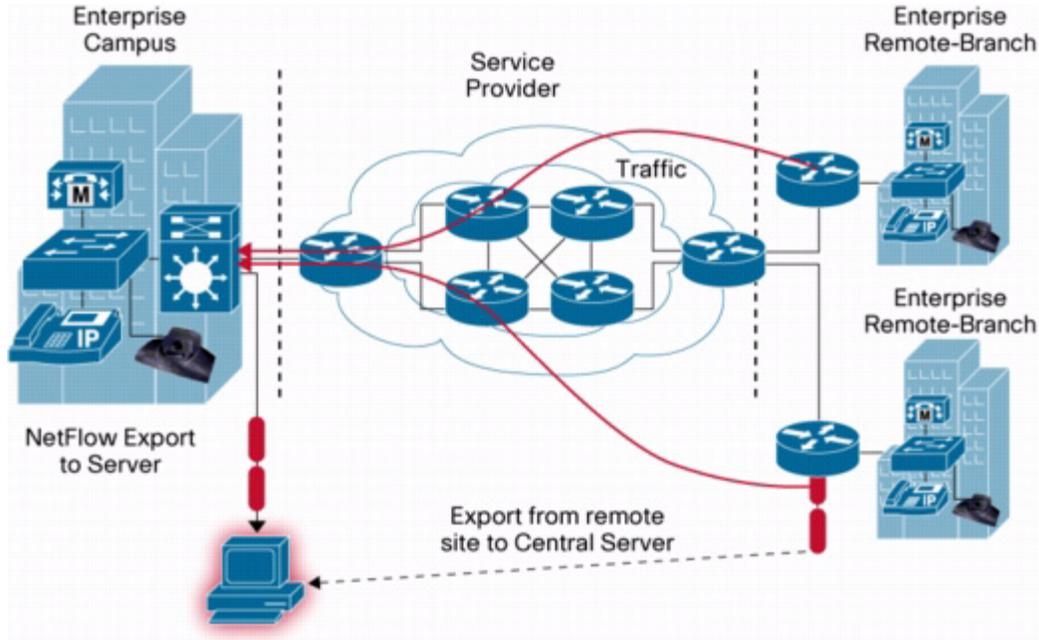
## NetFlow コレクタ サーバへのフローのエクスポート

タイマーは、フローが NetFlow コレクタ サーバにエクスポートされるタイミングを判断します。

次のいずれかが発生するときに、フローのエクスポートが可能になります。

- フローが一定時間非アクティブとなり、その間、フローで新しいパケットを受信していない場合。
- 長時間の FTP ダウンロードなど、フローがアクティブなタイマーより長く存在している場合。
- TCP フラグが、フローが終了したことを示している場合。つまり、FIN または RST フラグが表示されている場合です。
- フロー キャッシュが満杯で、一部の古いフローを削除して、新しいフローのための領域を作成する必要がある場合。

図 11-2 NetFlow コレクタ サーバへのフローのエクスポート



## NetFlow データの例

次の図は、NetFlow データの例を示しています。

図 11-3 NetFlow キャッシュの例

**1. Flow cache—The first unique packet creates a flow**

SrcIf	SrcPadd	DstIf	DstPadd	Protocol	TOS	Flgs	Pkts	Src Port	Src Msk	Src AS	Dst Port	Dst Msk	Dst AS	NextHop	Bytes/ Pkt	Active	Idle
Fa1/0	173.100.2.12	Fa0/0	10.0.227.12	11	80	10	11000	162	/24	5	163	/24	15	10.0.23.2	1528	1745	4
Fa1/0	173.100.3.2	Fa0/0	10.0.227.12	6	40	0	2491	15	/26	196	15	/24	15	10.0.23.2	740	41.5	1
Fa1/0	173.100.20.2	Fa0/0	10.0.227.12	11	80	10	10000	161	/24	180	10	/24	15	10.0.23.2	1428	1145.5	3
Fa1/0	173.100.6.2	Fa0/0	10.0.227.12	6	40	0	2210	19	/30	180	19	/24	15	10.0.23.2	1040	24.5	14

**2. Flow Aging Timers**

- Inactive Flow (15 sec is default)
- Long Flow (30 min (1800 sec) is default)
- Flow ends by RST or FIN TCP Flag

SrcIf	SrcPadd	DstIf	DstPadd	Protocol	TOS	Flgs	Pkts	Src Port	Src Msk	Src AS	Dst Port	Dst Msk	Dst AS	NextHop	Bytes/ Pkt	Active	Idle
Fa1/0	173.100.2.12	Fa0/0	10.0.227.12	11	80	10	11000	00A2	/24	5	00A2	/24	15	10.0.23.2	1528	1800	4

**3. Flows packaged in export packet**  
Non-aggregated Flows—Export Version 5 or 9

**4. Transport Flows to Reporting Server**

## ハイ アベイラビリティ

Cisco Nexus 1000V は、NetFlow のステートレス リスタートをサポートします。リブートまたはスーパーバイザ スイッチオーバーの後に、Cisco Nexus 1000V は実行コンフィギュレーションを適用します。

## NetFlow の前提条件

- NetFlow は追加メモリと CPU リソースを消費するため、リソース要件に注意する必要があります。
- メモリと CPU リソースは、フロー モニタ インターフェイスをホスティングする VEM が提供します。リソースは VEM 上の CPU コアの数によって、制限されています。

## 設定時の注意事項および制約事項

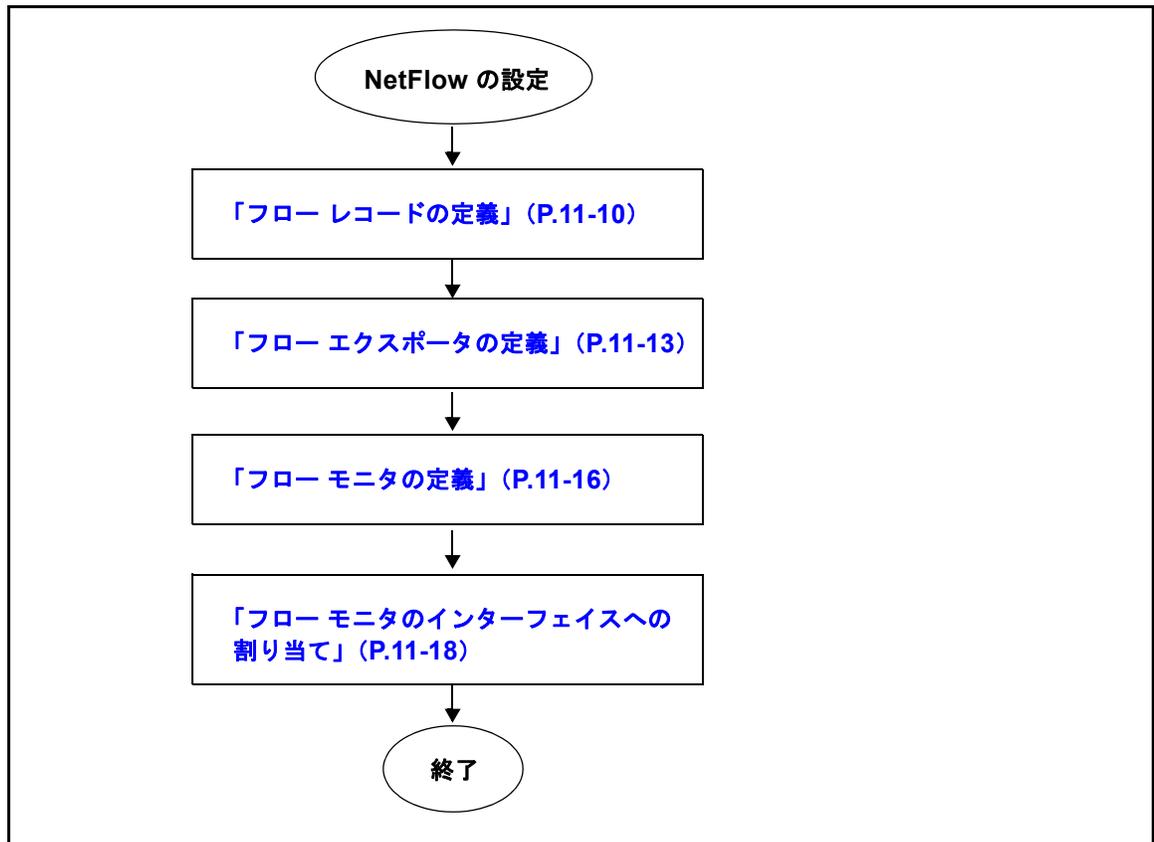
NetFlow に関する設定時の注意事項および制約事項は、次のとおりです。

- 送信元インターフェイスが設定されていない場合、NetFlow エクスポートはディセーブルの状態のままです。
- Cisco Nexus 1000V では、デフォルトで、エクスポートの送信元インターフェイスとして、Mgmt0 インターフェイスが設定されます。必要に応じて、送信元インターフェイスを変更できます。
- Cisco Nexus 1000V には、次のあらかじめ定義されたフロー レコードが含まれているため、新しいフロー レコードを作成する代わりに、これを使用できます。詳細については、「[フロー レコード定義](#)」(P.11-2) を参照してください。
  - netflow-original  
Cisco Nexus 1000V のあらかじめ定義された従来の、起点 AS を含む IPv4 入力 NetFlow
  - netflow ipv4 original-input  
Cisco Nexus 1000V のあらかじめ定義された従来の IPv4 入力 NetFlow
  - netflow ipv4 original-output  
Cisco Nexus 1000V のあらかじめ定義された従来の IPv4 出力 NetFlow
  - netflow ipv4 protocol-port  
Cisco Nexus 1000V のあらかじめ定義されたプロトコルおよびポート集約方式
- 最大 1 つのフロー モニタを、各方向の各インターフェイスに対して許可します。
- 最大 2 つのフロー エクスポートを、各モニタに対して許可します。

# NetFlow の設定

次のフローチャートは、NetFlow コンフィギュレーションプロセスを案内するために作成されています。各手順を完了したらこのフローチャートに戻り、すべての必要な手順を正しい順序で完了したことを確認してください。

フローチャート：NetFlow の設定



## フローレコードの定義

この手順を使用して、フローレコードを作成します。



(注)

任意で、「[フローレコード定義](#)」(P.11-2) で説明されている Cisco Nexus 1000V のあらかじめ定義されたレコードを使用できます。「[フローモニタの定義](#)」(P.11-16) を参照して、あらかじめ定義されたレコードを、フローモニタに適用します。

### 始める前に

この手順を開始する前に、次の点を理解または実行しておく必要があります。

- このフローレコードが一致するオプションを確認します。
- このフローレコードが収集するオプションを確認します。

詳細については、「[フロー レコード定義](#)」(P.11-2) を参照してください。



(注)

**show flow record** コマンドによる出力結果には、次の行が表示されますが、基になるコマンドは現在 Cisco Nexus 1000V ではサポートされていません。これらのコマンドを使用しても、コンフィギュレーションには影響はありません。

```
collect routing source as
collect routing destination as
collect routing next-hop address ipv4
```

## 手順の概要

1. **config t**
2. **flow record name**
3. **description string**
4. **match {ip {protocol | tos} | ipv4 {destination address | source address} | transport {destination-port | source-port}}**
5. **collect {counter {bytes [long] | packets [long]} | timestamp sys-uptime | transport tcp flags}**
6. **show flow record [name]**
7. **copy running-config startup-config**

## 手順の詳細

	コマンド	目的
ステップ 1	<b>config t</b>  例： n1000v# config t n1000v(config)#	CLI 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	<b>flow record name</b>  例： n1000v(config)# flow record RecordTest n1000v(config-flow-record)#	名前ごとにフロー レコードを作成し、特定のレコードで、CLI フロー レコード コンフィギュレーション モードに切り替えます。
ステップ 3	<b>description string</b>  例： n1000v(config-flow-record)# description Ipv4Flow	(任意) 63 文字以内の説明をこのフロー レコードに追加し、実行コンフィギュレーションに保存します。

コマンド	目的
<p>ステップ4 <b>match {ip {protocol   tos}   ipv4 {destination address   source address}   transport {destination-port   source-port}}</b></p> <p>例：  <pre>n1000v(config-flow-record)# match ipv4 destination address</pre> </p>	<p>次のいずれかに一致するフロー レコードを定義し、実行コンフィギュレーションに保存します。</p> <ul style="list-style-type: none"> <li>• <b>ip</b> : 次の IP オプションのいずれかと一致します。 <ul style="list-style-type: none"> <li>- protocol</li> <li>- tos (タイプ オブ サービス)</li> </ul> </li> <li>• <b>ipv4</b> : 次の ipv4 アドレス オプションのいずれかと一致します。 <ul style="list-style-type: none"> <li>- 送信元アドレス</li> <li>- 宛先アドレス</li> </ul> </li> <li>• <b>transport</b> : 次のトランスポート オプションのいずれかと一致します。 <ul style="list-style-type: none"> <li>- 宛先ポート</li> <li>- 送信元ポート</li> </ul> </li> </ul>
<p>ステップ5 <b>collect {counter {bytes [long]   packets [long]}   timestamp sys-uptime   transport tcp flags}</b></p> <p>例：  <pre>n1000v(config-flow-record)# collect counter packets</pre> </p>	<p>収集オプションを指定して、フロー レコードで収集する情報を定義し、実行コンフィギュレーションに保存します。</p> <ul style="list-style-type: none"> <li>• <b>counter</b> : 次のフォーマットのいずれかで、フロー レコード情報を収集します。 <ul style="list-style-type: none"> <li>- <b>bytes</b> : 32 ビットのカウンタで収集されます (ロングの 64 ビット カウンタが指定されている場合を除く)。</li> <li>- <b>packets</b> : 32 ビットのカウンタで収集されません (ロングの 64 ビット カウンタが指定されている場合を除く)。</li> </ul> </li> <li>• <b>timestamp sys-uptime</b> : フローの先頭または最終パケットに関するシステム稼働時間を収集します。</li> <li>• <b>transport tcp flags</b> : フローのパケットの TCP トランスポート レイヤ フラグを収集します。</li> </ul>
<p>ステップ6 <b>show flow record [name]</b></p> <p>例：  <pre>n1000v(config-flow-exporter)# show flow record RecordTest</pre> </p>	<p>(任意) フロー レコード情報を表示します。</p>
<p>ステップ7 <b>copy running-config startup-config</b></p> <p>例：  <pre>n1000v(config-flow-exporter)# copy running-config startup-config</pre> </p>	<p>(任意) 実行中のコンフィギュレーションをスタートアップ コンフィギュレーションにコピーし、リブートと再起動を行って、永久的に保存します。</p>

次に、フロー レコードを作成する例を示します。

```
n1000v# config t
n1000v(config)# flow record RecordTest
```

```

n1000v(config-flow-record)# description Ipv4flow
n1000v(config-flow-record)# match ipv4 destination address
n1000v(config-flow-record)# collect counter packets
n1000v(config-flow-record)# show flow record RecordTest
Flow record RecordTest:
  Description: Ipv4flow
  No. of users: 0
  Template ID: 0
  Fields:
    match ipv4 destination address
    match interface input
    match interface output
    match flow direction
    collect counter packets
n1000v(config-flow-record)#

```

## フロー エクスポートの定義

この手順を使用して、フロー レコードが NetFlow コレクタ サーバにエクスポートされる場所と方法を定義する、フロー エクスポートを作成します。

### 始める前に

この手順を開始する前に、次の点を理解または実行しておく必要があります。

- 最大 2 つのフロー エクスポートを、各モニタに対して許可します。
- NetFlow コレクタ サーバの宛先 IP アドレスを確認します。
- フロー レコードが送信される送信元インターフェイスを確認します。
- コレクタが待機しているトランスポート UDP を確認します。
- エクスポート フォーマット バージョン 9 がサポートされているバージョンです。

### 手順の概要

1. `config t`
2. `flow exporter name`
3. `description string`
4. `destination {ipv4-address | ipv6-address}`
5. `dscp value`
6. `source mgmt 0 slot/port`
7. `transport udp number`
8. `version 9`
9. `option {exporter-stats | interface-table} timeout seconds`
10. `template data timeout seconds`
11. `show flow exporter [name]`
12. `copy running-config startup-config`

## 手順の詳細

	コマンド	目的
ステップ 1	<b>config t</b>  例： n1000v# config t n1000v(config)#	CLI 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	<b>flow exporter name</b>  例： n1000v(config)# flow exporter ExportTest n1000v(config-flow-exporter)#	フロー エクスポートを作成して、実行コンフィギュレーションに保存します。次に、CLI フロー エクスポート コンフィギュレーション モードに切り替えます。
ステップ 3	<b>description string</b>  例： n1000v(config-flow-exporter)# description ExportV9	63 文字以内の説明をこのフロー エクスポートに追加し、実行コンフィギュレーションに保存します。
ステップ 4	<b>destination {ipv4-address   ipv6-address}</b>  例： n1000v(config-flow-exporter)# destination 192.0.2.1	このフロー エクスポートの宛先インターフェイスの IP アドレスを指定し、実行コンフィギュレーションに保存します。
ステップ 5	<b>dscp value</b>  例： n1000v(config-flow-exporter)# dscp 0	このフロー エクスポートの DiffServ コードポイント値を 0 ～ 63 の範囲で指定して、実行コンフィギュレーションに保存します。
ステップ 6	<b>source mgmt interface_number</b>  例： n1000v(config-flow-exporter)# source mgmt 0	フロー レコードが NetFlow コレクタ サーバに送信されるインターフェイスと数を指定して、実行コンフィギュレーションに保存します。
ステップ 7	<b>transport udp number</b>  例： n1000v(config-flow-exporter)# transport udp 200	NetFlow コレクションに到達するために使用する宛先 UDP ポートを、0 ～ 65535 の範囲で指定して、実行コンフィギュレーションに保存します。
ステップ 8	<b>version {9}</b>  例： n1000v(config-flow-exporter)# version 9 n1000v(config-flow-exporter-version-9)#	NetFlow エクスポート バージョン 9 を指定して、実行コンフィギュレーションに保存します。次に、エクスポート バージョン 9 コンフィギュレーション モードに切り替えます。

	コマンド	目的
ステップ 9	<pre>option {exporter-stats   interface-table   sampler-table} timeout value</pre> <p>例:</p> <pre>n1000v(config-flow-exporter-version-9)# option exporter-stats timeout 1200</pre>	<p>次のバージョン 9 のエクスポート再送タイマーと値のいずれかを、1 ~ 86400 秒の範囲で指定して、実行コンフィギュレーションに保存します。</p> <ul style="list-style-type: none"> <li>• <b>exporter-stats</b></li> <li>• <b>interface-table</b></li> <li>• <b>sampler-table</b></li> </ul>
ステップ 10	<pre>template data timeout seconds</pre> <p>例:</p> <pre>n1000v(config-flow-exporter-version-9)# template data timeout 1200</pre>	<p>テンプレート データ再送タイマーと値を、1 ~ 86400 秒の範囲で設定して、実行コンフィギュレーションに保存します。</p>
ステップ 11	<pre>show flow exporter [name]</pre> <p>例:</p> <pre>n1000v(config-flow-exporter)# show flow exporter</pre>	<p>(任意) フロー エクスポートに関する情報を表示します。</p>
ステップ 12	<pre>copy running-config startup-config</pre> <p>例:</p> <pre>n1000v(config-flow-exporter)# copy running-config startup-config</pre>	<p>(任意) 実行中のコンフィギュレーションをスタートアップコンフィギュレーションにコピーし、リブートと再起動を行って、永久的に保存します。</p>

次に、フロー エクスポートを作成する例を示します。

```
n1000v(config)# flow exporter ExportTest
n1000v(config-flow-exporter)# description ExportHamilton
n1000v(config-flow-exporter)# destination 192.0.2.1
n1000v(config-flow-exporter)# dscp 2
n1000v(config-flow-exporter)# source mgmt 0
n1000v(config-flow-exporter)# transport udp 200
n1000v(config-flow-exporter)# version 9
n1000v(config-flow-exporter-version-9)# option exporter-stats timeout 1200
n1000v(config-flow-exporter-version-9)# template data timeout 1200
n1000v(config-flow-exporter-version-9)# show flow exporter ExportTest
Flow exporter ExportTest:
  Description: ExportHamilton
  Destination: 192.0.2.1
  VRF: default (1)
  Destination UDP Port 200
  Source Interface Mgmt0
  DSCP 2
  Export Version 9
    Exporter-stats timeout 1200 seconds
    Data template timeout 1200 seconds
  Exporter Statistics
    Number of Flow Records Exported 0
    Number of Templates Exported 0
    Number of Export Packets Sent 0
    Number of Export Bytes Sent 0
    Number of Destination Unreachable Events 0
    Number of No Buffer Events 0
    Number of Packets Dropped (No Route to Host) 0
    Number of Packets Dropped (other) 0
```

```

Number of Packets Dropped (LC to RP Error) 0
Number of Packets Dropped (Output Drops) 1
Time statistics were last cleared: Never
n1000v(config-flow-exporter-version-9)#

```

## フロー モニタの定義

この手順を使用して、フロー モニタを作成し、フロー レコードとフロー エクスポートを作成したフロー モニタに関連付けます。

### 始める前に

- 最大 1 つのフロー モニタを、各方向の各インターフェイスに対して許可します。
- このフロー モニタを関連付ける既存のフロー エクスポート名を確認します。
- このフロー モニタを関連付ける既存のフロー レコード名を確認します。以前に作成したフロー レコードまたは次の Cisco Nexus 1000V のあらかじめ定義されたフロー レコードのいずれかを使用できます。
  - netflow-original
  - netflow ipv4 original-input
  - netflow ipv4 original-output
  - netflow ipv4 protocol-port

フロー レコードの詳細については、「[フロー レコード定義](#)」(P.11-2) を参照してください。

### 手順の概要

1. **config t**
2. **flow monitor *name***
3. **description *string***
4. **exporter *name***
5. **record *name***
6. **timeout {*active value* | *inactive value*}**
7. **cache {*size value*}**
8. **show flow monitor [*name*]**
9. **copy running-config startup-config**

## 手順の詳細

	コマンド	目的
ステップ 1	<code>config t</code>  例： n1000v# config t n1000v(config)#	CLI 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	<code>flow monitor name</code>  例： n1000v(config)# flow monitor MonitorTest n1000v(config-flow-monitor)#	名前ごとにフロー モニタを作成し、実行コンフィギュレーションに保存します。次に、CLI フロー モニタ コンフィギュレーション モードに切り替えます。
ステップ 3	<code>description string</code>  例： n1000v(config-flow-monitor)# description Ipv4Monitor	(任意) 指定されたフロー モニタの場合は、63 文字以下の英数字で説明用の文字列を追加し、実行コンフィギュレーションに保存します。
ステップ 4	<code>exporter name</code>  例： n1000v(config-flow-monitor)# exporter Exportv9	指定されたフロー モニタの場合は、既存のフロー エクスポートを追加し、実行コンフィギュレーションに保存します。
ステップ 5	<code>record {name   netflow {ipv4}}</code>  Example using Cisco Nexus 1000V pre-defined record: n1000v(config-flow-monitor)# record netflow-original  Example using user-defined record: n1000v(config-flow-monitor)# record RecordTest	指定されたフロー モニタの場合は、既存のフロー レコードを追加し、実行コンフィギュレーションに保存します。  <ul style="list-style-type: none"> <li><b>name</b> : 以前に作成したフロー レコード名、またはシスコが提供しているあらかじめ定義されたフロー レコード名。</li> <li><b>netflow</b> : 従来の NetFlow 収集方式 <ul style="list-style-type: none"> <li><b>ipv4</b> : 従来の IPv4 NetFlow 収集方式</li> </ul> </li> </ul>
ステップ 6	<code>timeout {active value   inactive value}</code>  例： n1000v(config-flow-monitor)# timeout inactive 600	(任意) 指定されたフロー モニタの場合、古いタイマーとキャッシュからの古いエントリの値を指定し、実行コンフィギュレーションに保存します。  <ul style="list-style-type: none"> <li><b>active</b> : アクティブ、ロング、またはタイムアウト。許可されている値は、60 ~ 4092 秒の範囲です。デフォルト値は 1800 秒です。</li> <li><b>inactive</b> : 非アクティブまたは通常のタイムアウト。許可されている値は、15 ~ 4092 秒の範囲です。デフォルト値は 15 秒です。</li> </ul>
ステップ 7	<code>cache {size value}</code>  例： n1000v(config-flow-monitor)# cache size 15000	(任意) 指定されたフロー モニタの場合、キャッシュサイズを 256 ~ 65536 エントリの範囲で指定し、実行コンフィギュレーションに保存します。  (注) このオプションは、モニタ キャッシュのメモリとパフォーマンスへの影響を抑えます。

	コマンド	目的
ステップ 8	<pre>show flow monitor [name]</pre> <p>例 :</p> <pre>n1000v(config-flow-monitor)# show flow monitor Monitor Test</pre>	(任意) 既存のフロー モニタに関する情報を表示します。
ステップ 9	<pre>copy running-config startup-config</pre> <p>例 :</p> <pre>n1000v(config-flow-monitor)# copy running-config startup-config</pre>	(任意) 実行中のコンフィギュレーションをスタートアップ コンフィギュレーションにコピーし、リブートと再起動を行って、永久的に保存します。

次に、フロー エクスポートを作成する例を示します。

```
n1000v(config)# flow monitor MonitorTest
n1000v(config-flow-monitor)# description Ipv4Monitor
n1000v(config-flow-monitor)# exporter ExportTest
n1000v(config-flow-monitor)# record RecordTest
n1000v(config-flow-monitor)# cache size 15000
n1000v(config-flow-monitor)# timeout inactive 600
n1000v(config-flow-monitor)# show flow monitor MonitorTest
Flow Monitor monitortest:
  Use count: 0
  Inactive timeout: 600
  Active timeout: 1800
  Cache Size: 15000
n1000v(config-flow-monitor)#
```

## フロー モニタのインターフェイスへの割り当て

この手順を使用して、フロー モニタをインターフェイスに割り当てます。

### 始める前に

- インターフェイスに使用するフロー モニタ名を確認します。
- インターフェイス タイプと数を確認します。

### 手順の概要

1. **config t**
2. **interface interface-type interface-number**
3. **ip flow monitor name {input | output}**
4. **show flow interface-type interface-number**
5. **copy running-config startup-config**

## 手順の詳細

	コマンド	目的
ステップ 1	<code>config t</code>  例： n1000v# config t n1000v(config)#	CLI 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	<code>interface interface-type</code> <code>interface-number</code>  例： n1000v(config)# interface veth 2 n1000v(config-if)#	指定されたインターフェイスの CLI インターフェイス コンフィギュレーション モードに切り替えます。
ステップ 3	<code>ip flow monitor name {input   output}</code>  例： n1000v(config-if)# ip flow monitor MonitorTest output	指定されたインターフェイスの場合は、入出力パケット用にフロー モニタを割り当て、実行コンフィギュレーションに保存します。
ステップ 4	<code>show flow interface-type</code> <code>interface-number</code>  例： n1000v(config-if)# show flow interface veth 2	(任意) 指定されたインターフェイスの場合、NetFlow 設定が表示されます。
ステップ 5	<code>copy running-config startup-config</code>  例： n1000v(config-if)# copy running-config startup-config	(任意) 実行中のコンフィギュレーションをスタートアップ コンフィギュレーションにコピーし、リブートと再起動を行って、永久的に保存します。

次に、フロー モニタをインターフェイスに割り当てる例を示します。

```
n1000v(config)# interface veth 2
n1000v(config-if)# ip flow monitor MonitorTest output
n1000v(config-if)# show flow interface veth 2
Interface veth 2:
  Monitor: MonitorTest
  Direction: Output
n1000v(config-if)#
```

## NetFlow の設定例

**例 11-5** この例では、新しいフロー レコードを使用して、フロー モニタを定義し、それをインターフェイスに適用します。

```
n1000v# config t
n1000v(config)# flow record RecordTest
n1000v(config-flow-record)# description Ipv4flow
```

```

n1000v(config-flow-record)# match ipv4 destination address
n1000v(config-flow-record)# collect counter packets
n1000v(config-flow-record)# exit
n1000v(config)# flow exporter ExportTest
n1000v(config-flow-exporter)# description ExportHamilton
n1000v(config-flow-exporter)# destination 192.0.2.1
n1000v(config-flow-exporter)# dscp 2
n1000v(config-flow-exporter)# source mgmt 0
n1000v(config-flow-exporter)# transport udp 200
n1000v(config-flow-exporter)# version 9
n1000v(config-flow-exporter-version-9)# option exporter-stats timeout 1200
n1000v(config-flow-exporter-version-9)# template data timeout 1200
n1000v(config-flow-exporter-version-9)# exit
n1000v(config-flow-exporter)# exit
n1000v(config)# flow monitor MonitorTest
n1000v(config-flow-monitor)# description Ipv4Monitor
n1000v(config-flow-monitor)# exporter ExportTest
n1000v(config-flow-monitor)# record RecordTest
n1000v(config-flow-monitor)# exit
n1000v(config)# interface veth 2/1
n1000v(config-if)# ip flow monitor MonitorTest output
n1000v(config-if)# show flow interface veth 2/1
Interface veth 2/1:
    Monitor: MonitorTest
    Direction: Output
n1000v(config-if)#

```

**例 11-6**           この例では、あらかじめ定義されたレコードを使用して、フロー モニタを定義し、それをインターフェイスに適用します。

```

n1000v# config t
n1000v(config)# flow exporter ExportTest
n1000v(config-flow-exporter)# description ExportHamilton
n1000v(config-flow-exporter)# destination 192.0.2.1
n1000v(config-flow-exporter)# dscp 2
n1000v(config-flow-exporter)# source mgmt 0
n1000v(config-flow-exporter)# transport udp 200
n1000v(config-flow-exporter)# version 9
n1000v(config-flow-exporter-version-9)# option exporter-stats timeout 1200
n1000v(config-flow-exporter-version-9)# template data timeout 1200
n1000v(config-flow-exporter-version-9)# exit
n1000v(config-flow-exporter)# exit
n1000v(config)# flow monitor MonitorTest
n1000v(config-flow-monitor)# description Ipv4Monitor
n1000v(config-flow-monitor)# exporter ExportTest
n1000v(config-flow-monitor)# record netflow-original
n1000v(config-flow-monitor)# exit
n1000v(config)# interface veth 2/1
n1000v(config-if)# ip flow monitor MonitorTest output
n1000v(config-if)# show flow interface veth 2/1
Interface veth 2/1:
    Monitor: MonitorTest
    Direction: Output
n1000v(config-if)#

```

# NetFlow の設定確認

NetFlow 設定を確認するには、表 11-1 で説明されているコマンドを使用します。

表 11-1 NetFlow の設定確認

コマンド	目的
<code>show flow exporter [name]</code>	NetFlow のフロー エクスポート マップ情報を表示します。 例 11-7 (P.11-21) を参照してください。
<code>show flow interface [interface-type number]</code>	NetFlow インターフェイスに関する情報を表示します。 例 11-8 (P.11-22) を参照してください。
<code>show flow monitor [name [cache module number [statistics module number]</code>	NetFlow フロー モニタに関する情報を表示します。 <b>(注)</b> <code>show flow monitor cache</code> コマンドは、キャッシュ エントリも表示するという点で、 <code>show flow monitor statistics</code> コマンドとは異なります。各プロセッサには独自のキャッシュがあるため、これらのコマンドのすべての出力内容は、サーバ (モジュールまたはホストとも呼ばれます) のプロセッサ数に基づいています。複数のプロセッサを使用して、1 つのフローのパケットを処理している場合、同じフローが各プロセッサに表示されます。 次の例を参照してください。 <ul style="list-style-type: none"> <li>「<a href="#">Show flow monitor</a>」 (P.11-22)</li> <li>「<a href="#">Show flow monitor cache module</a>」 (P.11-22)</li> <li>「<a href="#">Show flow monitor statistics module</a>」 (P.11-23)</li> </ul>
<code>show flow record [name]</code>	NetFlow のフロー レコード情報を表示します。

## 例 11-7 Show flow exporter

```
n1000v(config-flow-exporter-version-9)# show flow exporter ExportTest
Flow exporter ExportTest:
  Description: ExportHamilton
  Destination: 192.0.2.1
  VRF: default (1)
  Destination UDP Port 200
  Source Interface 2
  DSCP 2
  Export Version 9
  Exporter-stats timeout 1200 seconds
  Data template timeout 1200 seconds
  Exporter Statistics
    Number of Flow Records Exported 0
    Number of Templates Exported 0
```

```

Number of Export Packets Sent 0
Number of Export Bytes Sent 0
Number of Destination Unreachable Events 0
Number of No Buffer Events 0
Number of Packets Dropped (No Route to Host) 0
Number of Packets Dropped (other) 0
Number of Packets Dropped (IC to RP Error) 0
Number of Packets Dropped (Output Drops) 1
Time statistics were last cleared: Never
n1000v(config-flow-exporter-version-9)#

```

**例 11-8 Show flow interface**

```

n1000v(config-if)# show flow interface VEth2
Interface veth2:
  Monitor: MonitorTest
  Direction: Output
n1000v(config-if)#

```

**例 11-9 Show flow monitor**

```

n1000v(config)# show flow monitor
Flow Monitor MonitorTest:
  Description: Ipv4Monitor
  Use count: 1
  Flow Record: test
  Flow Exporter: ExportTest
  Inactive timeout: 15
  Active timeout: 1800
  Cache Size: 15000
Flow Monitor MonitorIpv4:
  Description: exit
  Use count: 70
  Flow Record: RecordTest
  Flow Exporter: ExportIpv4
  Inactive timeout: 15
  Active timeout: 1800
  Cache Size: 4096
n1000v(config)#

```

**例 11-10 Show flow monitor cache module**

```

n1000v# show flow monitor test_mon cache module 5
Cache type: Normal
Cache size (per-processor): 4096
High Watermark: 2
Flows added: 102
Flows aged: 099
  - Active timeout 0
  - Inactive timeout 099
  - Event aged 0
  - Watermark aged 0
  - Emergency aged 0
  - Permanent 0
  - Immediate aged 0
  - Fast aged 0

Cache entries on Processor0
  - Active Flows: 2
  - Free Flows: 4094

```

```

IPV4 SRC ADDR   IPV4 DST ADDR  IP PROT          INTF INPUT          INTF OUTPUT  FLOW DIRN
=====
0.0.0.0         255.255.255.255  17              Veth1              Input
7.192.192.10    7.192.192.2    1              Veth1              Eth5/2       Input
    
```

```

Cache entries on Processor1
- Active Flows:          0
- Free Flows:           4096
    
```

```

Cache entries on Processor2
- Active Flows:          1
- Free Flows:           4095
    
```

```

IPV4 SRC ADDR   IPV4 DST ADDR  IP PROT          INTF INPUT          INTF OUTPUT  FLOW DIRN
=====
7.192.192.10    7.192.192.1    1              Veth1              Eth5/2       Input
    
```

```

Cache entries on Processor3
- Active Flows:          0
- Free Flows:           4096
    
```

```

Cache entries on Processor4
- Active Flows:          0
- Free Flows:           4096
    
```

```

Cache entries on Processor5
- Active Flows:          0
- Free Flows:           4096
    
```

```

Cache entries on Processor6
- Active Flows:          0
- Free Flows:           4096
    
```

```

Cache entries on Processor7
- Active Flows:          0
- Free Flows:           4096
    
```

**例 11-11 Show flow monitor statistics module**

```

NX-1000v# show flow monitor test_mon statistics module 5
Cache type:                Normal
Cache size (per-processor): 4096
High Watermark:            2
Flows added:                105
Flows aged:
- Active timeout           0
- Inactive timeout         103
- Event aged               0
- Watermark aged           0
- Emergency aged           0
- Permanent                0
- Immediate aged           0
- Fast aged                0

Cache entries on Processor0
- Active Flows:            0
- Free Flows:              4096

Cache entries on Processor1
- Active Flows:            1
- Free Flows:              4095
    
```

```

Cache entries on Processor2
- Active Flows:          1
- Free Flows:           4095

Cache entries on Processor3
- Active Flows:          0
- Free Flows:           4096

Cache entries on Processor4
- Active Flows:          0
- Free Flows:           4096

Cache entries on Processor5
- Active Flows:          0
- Free Flows:           4096

Cache entries on Processor6
- Active Flows:          0
- Free Flows:           4096

Cache entries on Processor7
- Active Flows:          0
- Free Flows:           4096

```

**例 11-12 Show flow record**

```

n1000v(config-flow-record)# show flow record RecordTest
Flow record RecordTest:
  Description: Ipv4flow
  No. of users: 0
  Template ID: 0
  Fields:
    match ipv4 destination address
    match interface input
    match interface output
    match flow direction
    collect counter packets
n1000v(config-flow-record)#

```

## デフォルト設定

表 11-2 に、NetFlow パラメータのデフォルト設定をリスト表示します。

**表 11-2 デフォルトの NetFlow パラメータ**

パラメータ	デフォルト
source interface	mgmt0
match	方向とインターフェイス（着信/発信）
flow monitor active timeout	1800
flow monitor inactive timeout	15
flow monitor cache size	4096

## その他の関連資料

NetFlow の実装に関連する詳細情報については、次の項を参照してください。

- 「関連資料」 (P.11-25)
- 「標準規格」 (P.11-25)

## 関連資料

関連項目	マニュアル タイトル
Cisco NetFlow の概要	<a href="http://cisco.com/en/US/products/ps6601/products_ios_protocol_group_home.html">http://cisco.com/en/US/products/ps6601/products_ios_protocol_group_home.html</a>

## 標準規格

標準規格	タイトル
この機能でサポートされる新規または改訂された標準規格はありません。また、この機能による既存の標準規格サポートの変更はありません。	—

