トラフィック ミラーリングの設定

このモジュールでは、トラフィック ミラーリング機能の設定について説明します。トラフィック ミラーリングは、ポート ミラーリング、またはスイッチド ポート アナライザ（SPAN）と呼ばれます。

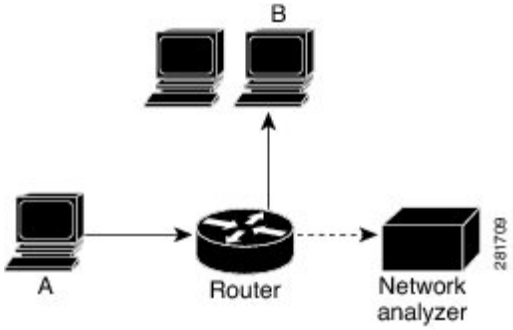
トラフィック ミラーリングの概要

トラフィック ミラーリングは、ポート ミラーリングまたはスイッチド ポート アナライザ（SPAN）と呼ばれることもある、シスコ独自の機能です。この機能を利用すると、一連のポートに入ってくる、または出ていくネットワーク トラフィックをモニタすることができます。このトラフィックを同じルータ上の宛先ポートに渡すことができます。

トラフィック ミラーリングでは、1 つまたは複数の送信元ポートからのトラフィックをコピーし、コピーされたトラフィックを 1 つまたは複数の宛先に送信してネットワーク アナライザまたはその他のモニタリング デバイスに分析させます。トラフィック ミラーリングは、送信元インターフェイスまたはサブインターフェイス上のトラフィックのフローに影響を与えず、ミラーリングされたトラフィックは宛先インターフェイスまたはサブインターフェイスに送信されます。

たとえば、トラフィック アナライザをルータに接続してホスト A によってホスト B に送信されるイーサネット トラフィックをキャプチャできます。

図 *1:* トラフィック ミラーリング動作



ローカル トラフィック ミラーリングが有効になっている場合、ホスト A から送信されるすべてのパケットのコピーを受信するように設定されたポートに、トラフィック アナライザを直接接続します。このポートを「トラフィック ミラーリング ポート」といいます。このマニュアルの他の項で、この機能を調整する方法について説明します。

トラフィック ミラーリングのタイプ

次のタイプのトラフィック ミラーリングがサポートされています。

- **ローカル トラフィック ミラーリング**：最も基本的な形式のトラフィック ミラーリングです。ネットワーク アナライザまたはスニファは宛先インターフェイスに直接接続します。つまり、すべてのモニタ対象ポートが宛先ポートと同じルータ上に存在します。
- **リモート トラフィック ミラーリング**：IP ネットワークを介し、GRE トンネルを通じてネットワーク アナライザに到達できます。

（注） ethernet キーワードを設定した場合、それぞれのパケットのコピーにはレイヤ 2 ヘッダーが含まれています。これはミラーリングされたパケットがルーティングできないことを示しているため、GRE トンネルの末端をネットワーク アナライザにする必要があります。

- **ACL ベースのトラフィック ミラーリング**：トラフィックはインターフェイス ACL の設定に基づいてミラーリングされます。

  インターフェイス アクセス リストの定義に基づいてトラフィックをミラーリングできます。レイヤ 3 トラフィックをミラーリングする際は、**ipv4 access-list** コマンドまたは **ipv6 access-list** コマンド使用し、**capture** オプションを指定して ACL を設定します。**permit** コマンドと **deny** コマンドによって、通常のトラフィックの動作を決定します。**capture** オプションは、パケットが宛先ポートにミラーリングされることを指定します。このオプションは許可タイプのアクセス制御エントリ（ACE）でのみサポートされています。

（注） リリース 6.5.1 より前では、ACL ベースのトラフィック ミラーリングには UDK（ユーザ定義の TCAM キー）と **enable-capture** オプションを使用して **capture** オプションを ACL に設定できるようにする必要がありました。

ERSPAN

Encapsulated Remote Switched Port Analyzer（ERSPAN）は、IP ネットワークでミラーリングされたトラフィックを転送します。トラフィックは、送信元ルータでカプセル化され、ネットワーク間を転送されます。パケットは宛先ルータでカプセル化解除され、宛先インターフェイスに送信されます。

ERSPAN は、GRE トンネルを通過してリモート サイトに送信されるトラフィックのミラーリングを実行します。モニタ セッションの宛先として使用する GRE トンネルの設定の詳細については、「*GRE* トンネルの設定」の章を参照してください。

制約事項

ACL を使用したトラフィックのミラーリングには次の一般的な制約事項が適用されます。

- トラフィック ミラーリング カウンタはサポートされていません。
- ACL ベースのトラフィック ミラーリングはレイヤ 2（イーサネットサービス）ACL ではサポートされていません。
- トラフィックのデフォルトのミラーリングを回避するために、送信元インターフェイス上、または送信元インターフェイスと同じネットワーク処理ユニット上の任意のインターフェイス上に ACL を設定します。バンドル インターフェイスが送信元インターフェイスの場合は、アクティブなすべてのバンドル メンバーと同じネットワーク処理ユニットの任意のインターフェイス上に ACL を設定します。バンドル メンバーは、複数の NPU 上に配置できます。また、設定した ACL が SPAN 設定と同じプロトコル タイプと方向であることを確認します。たとえば、IPv4 または IPv6 の ACL を使用して SPAN を設定する場合は、そのネットワーク処理ユニットに入力 IPv4 ACL または IPv6 ACL をそれぞれ設定します。

次の一般的な制限が SPAN に適用されます。

- SPAN はポートレベルの送信元インターフェイスのみをサポートしています。

次の一般的な制約事項が ERSPAN と SPAN ACL に適用されます。

- ERSPAN トンネルの統計情報はサポートされていません。
- SPAN カウンタはサポートされていせん。

- SPAN 機能と ER-SPAN 機能の両方を同時にルータ上に設定することはできません。SPAN 機能または ERSPAN 機能のいずれかを同じルータ上で設定できます。
- ERSPAN セッション ID の値は常にゼロです。
  - ERPAN を設定するための IOS XR コマンドは使用できません。
- ERSPAN のネクストホップには解決された ARP が必要です。
  - その他のトラフィックまたはプロトコルで ARP をトリガーします。
- ERSPAN は MPLS を介して移動できません。
  - 追加ルータは MPLS でカプセル化される場合があります。
- ERSPAN のカプセル化解除はサポートされていません。
- GRE ネクスト ホップがサブインターフェイスを介して到達可能な場合、ERSPAN は機能しません。ERSPAN が機能するには、メイン インターフェイスを介してネクスト ホップに到達可能である必要があります。
- Rx 方向（入力方向 v4 ACL または v6 ACL）では SPAN-ACL のみがサポートされています。
- SPAN-ACL では、MPLS トラフィックをキャプチャできません。
  - MPLS トラフィックの ACL はサポートされていません。

トラフィック ミラーリングの設定方法

ここでは、トラフィック ミラーリングを設定する方法について説明します。

リモート トラフィック ミラーリングの設定

手順

ステップ 1 **configure**

例：

```
RP/0/RP0/cpu 0: router# configure
```

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 **monitor-session** *session-name*

例：

```
RP/0/RP0/cpu 0: router(config)# monitor-session mon1 ethernet
RP/0/RP0/cpu 0: router(config-mon)#
```

モニタ セッションを定義し、モニタ セッション コンフィギュレーション モードを開始します。

ステップ 3 **destination interface** *tunnel-ip*

例：

```
RP/0/RP0/cpu 0: router(config-mon)# destination interface tunnelip3
```

トラフィックを複製する宛先サブインターフェイスを指定します。

ステップ 4 **exit**

例：

```
RP/0/RP0/cpu 0: router(config-mon)# exit
RP/0/RP0/cpu 0: router(config)#
```

モニタ セッション 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。

ステップ 5 **interface** *type number*

例：

```
RP/0/RP0/cpu 0: router(config)# interface HundredGigE 0/0/1/0
```

指定した送信元インターフェイスに対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。インターフェイス番号は、*rack*/*slot*/*module*/*port* 表記で入力します。ルータの構文の詳細については、疑問符（?）を使用してオンライン ヘルプを参照してください。

ステップ 6 **monitor-session** *session-name* **ethernet direction rx-onlyport-only**

例：

```
RP/0/RP0/cpu 0: router(config-if)# monitor-session mon1 ethernet
direction rx-only port-only
```

このインターフェイスで使用されるモニタ セッションを指定します。**direction** キーワードを使用して、入力または出力のトラフィックのみをミラーリングすることを指定します。

ステップ 7 **end** または **commit**

例：

```
RP/0/RP0/cpu 0: router(config-if)# end
```

または

```
RP/0/RP0/cpu 0: router(config-if)# commit
```

設定変更を保存します。

- **end** コマンドを実行すると、次に示す変更のコミットを求めるプロンプトが表示されます。

```
Uncommitted changes found, commit them before exiting (yes/no/cancel)?
[cancel]:
```

- yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。

- no と入力すると、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。

- cancel と入力すると、ルータは現在のコンフィギュレーション セッションで継続されます。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。

- 実行コンフィギュレーションファイルに設定変更を保存し、コンフィギュレーションセッションを継続するには、**commit** コマンドを使用します。

実行コンフィギュレーションファイルに設定変更を保存し、コンフィギュレーションセッションを継続するには、**commit** コマンドを使用します。

ステップ 8 **show monitor-session [session-name] status [detail] [error]**

例：

```
RP/0/RP0/cpu 0: router# show monitor-session
```

トラフィック ミラーリングセッションに関する情報を表示します。

設定可能な送信元インターフェイスの接続

手順

ステップ 1 **configure**

例：

```
RP/0/RP0/cpu 0: router# configure
```

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 **interface** *type number*

例：

```
RP/0/RP0/cpu 0: router(config)# interface HundredGigE 0/0/1/0
```

指定した送信元インターフェイスに対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。インターフェイス番号は、*rack*/*slot*/*module*/*port* 表記で入力します。ルータの構文の詳細については、疑問符（?）を使用してオンライン ヘルプを参照してください。

ステップ 3 **ipv4 access-group** *acl-name* {**ingress** | **egress**}

例：

```
RP/0/RP0/cpu 0: router(config-if)# ipv4 access-group acl1 ingress
```

インターフェイスへのアクセスを制御します。

ステップ 4 **monitor-session** *session-name* **ethernet direction rx-onlyport-level acl**

例：

```
RP/0/RP0/cpu 0: router(config-if)# monitor-session mon1 ethernet direction rx-only
port-level acl
RP/0/RP0/cpu 0: router(config-if-mon)#
```

送信元インターフェイスにモニタ セッションを付加し、モニタ セッション コンフィギュレーション モードを開始します。

（注） **rx-only** は入力トラフィックのみが複製されることを指定します。

ステップ 5 **acl**

例：

```
RP/0/RP0/cpu 0: router(config-if-mon)# acl
```

定義された ACL に従ってトラフィックをミラーリングすることを指定します。

（注） ACL を名前で設定した場合は、それによってインターフェイス上で設定されている可能性がある ACL がオーバーライドされます。

ステップ 6 **exit**

例：

```
RP/0/RP0/cpu 0: router(config-if-mon)# exit
RP/0/RP0/cpu 0: router(config-if)#
```

モニタ セッション コンフィギュレーション モードを終了し、インターフェイス コンフィギュレーション モードに戻ります。

ステップ 7 **end** または **commit**

例：

```
RP/0/RP0/cpu 0: router(config-if)# end
```

または

```
RP/0/RP0/cpu 0: router(config-if)# commit
```

設定変更を保存します。

- **end** コマンドを実行すると、次に示す変更のコミットを求めるプロンプトが表示されます。

```
Uncommitted changes found, commit them before exiting (yes/no/cancel)?
[cancel]:
```

- yes と入力すると、実行コンフィギュレーション ファイルに変更が保存され、コンフィギュレーション セッションが終了して、ルータが EXEC モードに戻ります。

- no と入力すると、コンフィギュレーションセッションが終了して、ルータが EXEC モードに戻ります。変更はコミットされません。

- cancel と入力すると、ルータは現在のコンフィギュレーション セッションで継続されます。コンフィギュレーション セッションは終了せず、設定変更もコミットされません。

- 実行コンフィギュレーションファイルに設定変更を保存し、コンフィギュレーションセッションを継続するには、**commit** コマンドを使用します。

ステップ 8 **show monitor-session [session-name] status [detail] [error]**

例：

```
RP/0/RP0/cpu 0: router# show monitor-session status
```

モニタ セッションに関する情報を表示します。

トラフィック ミラーリングへの UDF ベースの ACL の設定

手順

	コマンドまたはアクション	目的
ステップ 1	**configure** 例： `RP/0/RP0/cpu 0: router# configure`	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	**udf** *udf-name* **header** {**inner** | **outer**} {**l2** | **l3** | **l4**} **offset** *offset-in-bytes* **length** *length-in-bytes* 例： `RP/0/RP0/cpu 0: router(config)# udf udf3 header outer l4 0 length 1 (config-mon)#`	個別の UDF 定義を設定します。UDF の名前、オフセット元のネットワーキング ヘッダー、抽出するデータの長さを指定できます。 **inner** キーワードまたは **outer** キーワードは、カプセル化されていないレイヤ 3 またはレイヤ 4 のヘッダーからのオフセットの開始を指定するか、またはカプ

	コマンドまたはアクション	目的

	例：
	`RP/0/RP0/cpu 0: router(config)# udf udf3 header inner l4 10 length 2 (config-mon)#`
	例：
	`RP/0/RP0/cpu 0: router(config)# udf udf3 header outer l4 50 length 1 (config-mon)#`
		セル化されたパケットがある場合は内部 L3/L4 からのオフセットの開始を指定します。
		（注） 任意のヘッダーの開始部分から許容される最大オフセットは 63 バイトです。
		length キーワードはオフセットからの長さをバイト単位で指定します。指定できる値の範囲は 1 ～ 4 です。

ステップ 3	**hw-module profile tcam format access-list** {**ipv4 | ipv6**} [*acl-qualifiers*] [**udf1** *udf-name1 ...* **udf8** *udf-name8*] **enable-capture** 例： `RP/0/RP0/cpu 0: router(config)# hw-module profile tcam format access-list ipv4 src-addr dst-addr src-port dst-port proto tcp-flags packet-length frag-bit udf1 udf-test1 udf2 udf-test2 enable-capture`	ハードウェアに送信される ACL キー定義にユーザ定義フィールドを追加します。 （注） 新しい TCAM プロファイルを有効にするには、ライン カードのリロードが必要です。
ステップ 4	**ipv4 access-list** *acl-name* 例： `RP/0/RP0/cpu 0: router(config))# ipv4 access-list acl1`	ACL を作成して、IP ACL コンフィギュレーション モードを開始します。 *acl-name* 引数の長さは最大 64 文字です。

ステップ 5	**permit** *regular-ace-match-criteria* **udf** *udf-name1 value1 ... udf-name8 value8*
	例：
	`RP/0/RP0/cpu 0: router(config-ipv4-acl)# 10 permit ipv4 any any udf udf1 0x1234 0xffff udf3 0x56 0xff capture`
	`RP/0/RP0/cpu 0: router(config-ipv4-acl)# 30 permit ipv4 any any dscp af11 udf udf5 0x22 0x22 capture`
		UDF と一致する ACL を設定します。

ステップ 6	**exit** 例： `RP/0/RP0/cpu 0: router(config-ipv4-acl)# exit`	IP ACL 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。

	コマンドまたはアクション	目的
ステップ 7	**interface***type number* 例： `RP/0/RP0/cpu 0: router(config)# interface HundredGigE 0/0/1/0`	インターフェイスを設定し、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。
ステップ 8	**ipv4 access-group** *acl-name* **ingress** 例： `RP/0/RP0/cpu 0: router(config-if)# ipv4 access-group acl1 ingress`	アクセス リストをインターフェイスに適用します。
ステップ 9	**commit** 例： `RP/0/RP0/cpu 0: router(config-if)# commit`	アクセス リストをインターフェイスに適用します。

トラフィック ミラーリングに関する追加情報

トラフィック ミラーリング用語

- 入力トラフィック：ルータに着信するトラフィック。
- 出力トラフィック：ルータから発信されるトラフィック。
- 送信元（SPAN）インターフェイス：SPAN機能を使用してモニタされているインターフェイス。
- 送信元ポート：トラフィック ミラーリングを使用してモニタされるポート。モニタ対象ポートとも呼ばれます。
- 宛先ポート：送信元ポートをモニタするポート。通常は、このポートにネットワーク アナライザが接続されます。「モニタリング ポート」とも呼ばれます。
- モニタ セッション：SPAN設定の集合に名前を付けたもの。この集合は宛先と送信元のインターフェイスで構成され、宛先は1つ、送信元は1つまたは複数となる可能性があります。

送信元ポートの特性

モニタ対象ポートとも呼ばれる送信元ポートは、ネットワーク トラフィック分析のためにモニタするルーテッドポートです。単一トラフィックのミラーリングセッションでは、送信元ポー

トのトラフィックをモニタできます。NCS 5500 シリーズ ルータは、最大 800 個の送信元ポートをサポートできます。

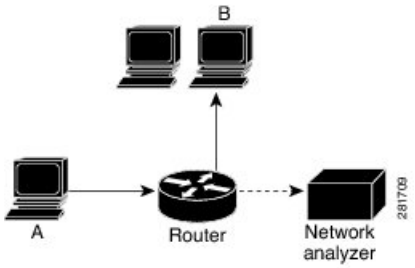
送信元ポートの特性は、次のとおりです。

- バンドル インターフェイス、100 ギガビット イーサネット、10 ギガビット イーサネットなどのデータ ポート タイプを使用できます。

（注） ブリッジ グループ仮想インターフェイス（BVI）はサポートされません。

- 各送信元ポートは、1 つのトラフィック ミラーリング セッションでのみモニタできます。
- ポートを送信元ポートとして使用した場合は、同じポートを宛先ポートとしては使用できません。
- 各送信元ポートはローカル トラフィック ミラーリングをモニタする方向（入力、出力、または両方）を指定して設定できます。リモート トラフィック ミラーリングは、入力方向と出力方向の両方でサポートされています。バンドルの場合は、モニタ方向はグループ内のすべての物理ポートに適用されます。

図 *2:* トラフィック ミラーリングを使用した *Cisco NCS 5500* ルータでのネットワーク分析



上の図では、ネットワーク アナライザが接続されるポートは、ホスト A から送信されるすべてのパケットのコピーを受信するように設定されています。このポートを「トラフィック ミラーリング ポート」といいます。

モニタ セッションの特性

モニタ セッションは、1 つの宛先インターフェイスと、場合によっては多くの送信元インターフェイスで構成されるトラフィック ミラーリング設定の集まりです。どのモニタ セッションでも、送信元インターフェイス（送信元ポートと呼ばれる）からのトラフィックは、モニタリング ポートまたは宛先ポートに送信されます。1 つのモニタリング セッションに複数の送信元ポートがある場合は、多数のミラーリングされたトラフィック ストリームからのトラフィックが宛先ポートにおいて結合されます。その結果、宛先ポートからのトラフィックは、1 つまたは複数の送信元ポートからのトラフィックの組み合わせになります。

モニタ セッションには次の特性があります。

- 1 台のルータで、最大 4 つのモニタ セッションを実行できます。
- 単一のモニタ セッションの宛先ポートは 1 つだけです。
- 1 つの宛先ポートは 1 つのモニタ セッションだけに属することができます。
- 1 つのモニタ セッションあたりの送信元ポートの最大数は 800 です。ただし、すべてのモニタリング セッションの送信元ポート数合計が 800 を超えないものとします。

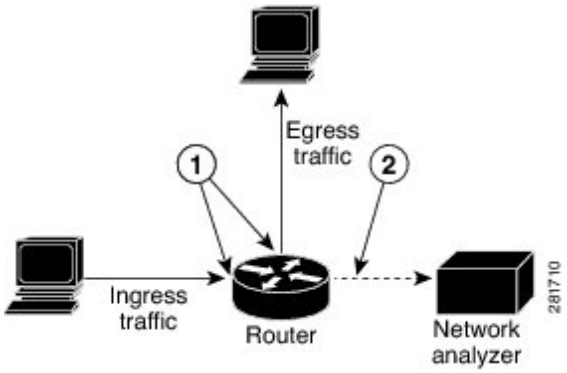
宛先ポートの特性

各セッションには、送信元ポートからのトラフィックのコピーを受信する宛先ポートが必要です。

宛先ポートの特性は、次のとおりです。

- 宛先ポートは、ローカル トラフィック ミラーリングを行う送信元ポートと同じルータ上にある必要があります。リモート ミラーリングの場合、宛先は常に GRE トンネルになります。
- ローカル ミラーリングの宛先ポートには、イーサネット物理ポート、EFP、GRE トンネル インターフェイスを使用できますが、バンドル インターフェイスは使用できません。有効なのはレイヤ 2 またはレイヤ 3 の転送インターフェイスです。
- NCS5500 の宛先ポートを vlan サブインターフェイスにすることはできません。
- いつでも、宛先ポートは 1 つのトラフィック ミラーリング セッションだけに参加できます。1 つのトラフィック ミラーリング セッションの宛先ポートは、別のトラフィック ミラーリング セッションの宛先ポートにできません。つまり、2 つのモニタ セッションの宛先ポートが同一であってはなりません。
- 宛先ポートは、送信元ポートにはできません。

図 *3:* トラフィック ミラーリングを使用した *Cisco NCS 5500* シリーズ ルータでのネットワーク分析



上の図のコールアウトは次を示しています。

1. 送信元トラフィック ミラーリング ポート（入力または出力のトラフィック ポート）。

2. 宛先トラフィック ミラーリングポート。

トラフィック ミラーリングの設定例

ここでは、トラフィック ミラーリングを設定する方法の例を示します。

物理インターフェイスを使用したトラフィックミラーリング（ローカル）：例

次に、物理インターフェイスを使用したトラフィック ミラーリングの基本設定の例を示します。

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# monitor-session ms1
RP/0/RP0/CPU0:router(config-mon)# destination interface HundredGigE0/0/1/0
RP/0/RP0/CPU0:router(config-mon)# commit

RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# interface TenGigE0/0/0/0
RP/0/RP0/CPU0:router(config-if)# monitor-session ms1 port-level direction rx-only
RP/0/RP0/CPU0:router(config-if)# commit
```

モニタ セッション ステータスの表示：例

次に、**status** キーワードを指定した **show monitor-session** コマンドの出力例を示します。

```
RP/0/RP0/CPU0:router# show monitor-session status

Monitor-session cisco-rtp1
Destination interface HundredGigE 0/0/1/0
================================================================================
Source Interface   Dir  Status
------------------- ---- ------------------------------------------------------
TenGigE0/0/0/4       Both Operational
TenGigE0/0/0/17      Both Operational

RP/0/RSP0/CPU0:router# show monitor-session status detail

Monitor-session sess1
 Destination interface is not configured
 Source Interfaces
 -----------------
 TenGigE0/0/0/0
  Direction: Both
  ACL match: Disabled
  Portion:  Full packet
  Status:  Not operational (destination interface not known).
 TenGigE0/0/0/1
  Direction: Both
  ACL match: Disabled
  Portion:  First 100 bytes
```

```
RP/0/RP0/CPU0:router# show monitor-session status error

Monitor-session ms1
Destination interface TenGigE0/0/0/15 is not configured
================================================================================
Source Interface   Dir  Status
------------------- ---- -------------------------------------------------

Monitor-session ms2
Destination interface is not configured
================================================================================
Source Interface   Dir  Status
------------------- ---- -------------------------------------------------


RP/0/RP0/CPU0:router# show monitor-session test status

Monitor-session test (ipv4)

Destination Nexthop 255.254.254.4
================================================================================================
Source Interface   Dir         Status
------------------------------------------------------------------------------------------------
Gi0/0/0/2.2     Rx    Not operational (source same as destination)
Gi0/0/0/2.3      Rx    Not operational (Destination not active)
Gi0/0/0/2.4      Rx    Operational
Gi0/0/0/4       Rx    Error: see detailed output for explanation

RP/0/RP0/CPU0:router# show monitor-session test status error

Monitor-session test
Destination Nexthop ipv4 address 255.254.254.4
================================================================
Source Interface     Status
----------------------------------------------------------------
Gi0/0/0/4    < Error: FULL Error Details >
```

トラフィック ミラーリングのトラブルシューティング

トラフィック ミラーリングに問題が発生した場合は、**show monitor-session status** コマンドの出力を確認することからトラブルシューティングを開始します。このコマンドは、すべてのセッションおよび送信元インターフェイスの記録された状態を表示します。

```
# show monitor-session status

Monitor-session ms1
<session status>
================================================================================


 Interface      Dir   Status
--------------------- ----  -------------------------------------------------
Gi0/1/0/0.10          Both  <Source interface status>
Gi0/1/0/0.11          Rx    <Source interface status>
Gi0/1/0/0.12          Tx    <Source interface status>
Gi0/2/0/0 (port)      Rx    <Source interface status>
```

上記の例では、<Session status> とマークされた行は、次のいずれかの設定エラーを示している可能性があります。

Session Status	説明
Session is not configured globally	グローバル設定にセッションが存在していません。**show run** コマンドの出力を確認し、セッションが正しい名前で設定されていることを確認します。
Destination interface <intf> (<down-state>)	宛先インターフェイスは、Interface Manager でアップ状態になっていません。**show interfaces** コマンドを使用して状態を確認できます。設定を調べて、インターフェイスがアップ状態にならない原因を特定します（たとえば、サブインターフェイスが適切なカプセル化の設定を必要としています）。

<Source interface status> は次のメッセージを報告できます。

Source Interface Status	説明
Operational	トラフィック ミラーリング PI において、すべてのものが正しく動作しているようです。ミラーリングが期待どおりに動作しない場合は、まずプラットフォーム チームと協力して調査します。
Not operational (Session is not configured globally)	グローバル設定にセッションが存在していません。**show run** コマンドの出力を確認し、セッションが正しい名前で設定されていることを確認します。
Not operational (destination not known)	セッションは存在していますが、宛先インターフェイスが設定されていないか、そのセッションに指定されている宛先インターフェイスが存在していません（たとえば、宛先がまだ作成されていないサブインターフェイスであるなど）。
Not operational (source same as destination)	セッションは存在していますが、宛先と送信元が同じインターフェイスであるため、トラフィック ミラーリングは機能しません。
Not operational (destination not active)	宛先インターフェイスまたは疑似配線がアップ状態ではありません。対応する *Session status* のエラーメッセージで、提案されている解決方法を確認します。

Source Interface Status	説明
Not operational (source state <down-state>)	送信元インターフェイスはアップ状態ではありません。**show interfaces** コマンドを使用して状態を確認できます。設定を調べて、インターフェイスがアップ状態にならない原因を特定します（たとえば、サブインターフェイスが適切なカプセル化の設定を必要としています）。
Error: see detailed output for explanation	トラフィック ミラーリングでエラーが発生しました。**show monitor-session status detail** コマンドを実行して詳細情報を表示します。

show monitor-session status detail コマンドは、設定パラメータの詳細情報と、検出されたエラー（ある場合）を表示します。次に例を示します。

RP/0/RP0/cpu 0: router **show monitor-session status detail**

```
Monitor-session sess1
 Destination interface is not configured
 Source Interfaces
 -----------------
 TenGigE0/0/0/1
  Direction: Both
  ACL match: Disabled
  Portion:  Full packet
  Status:  Not operational (destination interface not known)
 TenGigE0/0/0/2
  Direction: Both
  ACL match: Disabled
  Portion:  First 100 bytes
  Status: Not operational (destination interface not known). Error: 'Viking SPAN PD'
detected the 'warning' condition 'PRM connection
          creation failure'.
Monitor-session foo
 Destination next-hop TenGigE 0/0/0/0
 Source Interfaces
 -----------------
 TenGigE 0/0/0/1.100:
  Direction: Both
  Status:  Operating
 TenGigE 0/0/0/2.200:
  Direction: Tx
  Status:  Error: <blah>

Monitor session bar
 No destination configured
 Source Interfaces
 -----------------
 TenGigE 0/0/0/3.100:
  Direction: Rx
  Status:  Not operational(no destination)
```

次に追加のトレースとデバッグのコマンドを示します。

```
RP/0/RP0/cpu 0: router# show monitor-session platform trace ?

 all   Turn on all the trace
 errors Display errors
 events Display interesting events

RP/0/RP0/cpu 0: router# show monitor-session trace ?

 process Filter debug by process

RP/0/RP0/cpu 0: router# debug monitor-session platform ?

 all   Turn on all the debugs
 errors VKG SPAN EA errors
 event  VKG SPAN EA event
 info  VKG SPAN EA info

RP/0/RP0/cpu 0: router# debug monitor-session process all

RP/0/RP0/cpu 0: router# debug monitor-session process ea

RP/0/RP0/cpu 0: router# debug monitor-session process ma

RP/0/RP0/cpu 0: router# show monitor-session process mgr

 detail  Display detailed output
 errors  Display only attachments which have errors
 internal Display internal monitor-session information
 |     Output Modifiers

RP/0/RP0/cpu 0: router# show monitor-session status

RP/0/RP0/cpu 0: router# show monitor-session status errors

RP/0/RP0/cpu 0: router# show monitor-session status internal
```

UDF ベースの ACL の確認

show monitor-session status detail コマンドを使用して、ACL の UDF の設定を確認します。

```
RP/0/RP0/CPU0:leaf1# show monitor-session 1 status detail

Fri May 12 19:40:39.429 UTC
Monitor-session 1
  Destination interface tunnel-ip3
  Source Interfaces
  -----------------
  TenGigE0/0/0/15
    Direction:  Rx-only
    Port level: True
    ACL match:  Enabled
    Portion:    Full packet
    Interval:   Mirror all packets
    Status:     Not operational (destination not active)
```