



## 未加工データのフォーマッティング：RDR フォーマッタと NetFlow エクスポート

### 概要

Cisco Service Control には、収集したレポートデータの収集、集計、保存、および処理のために、次の 2 つのプロトコルを介して外部アプリケーションに配信するという役割もあります。

- RDRv1：Service Control 独自のエクスポート プロトコル
- NetFlow V9：業界標準のエクスポート プロトコル

この 2 つのプロトコルは、同じ配置内で同時に使用できます。ただし、データの送信先となる特定のいずれかの配信先（外部コレクタ）への送信に設定できるプロトコルは 1 つだけです。

- 「RDR フォーマッタと NetFlow エクスポート サポート」(P.9-1)
- 「データの送信先とカテゴリの設定」(P.9-6)
- 「RDR フォーマッタの設定」(P.9-12)
- 「NetFlow エクスポート サポートの設定」(P.9-13)
- 「RDR からカテゴリへのダイナミック マッピングの設定」(P.9-14)
- 「データ送信先の設定と統計情報の表示」(P.9-15)
- 「ラインカードによる RDR 送信のディセーブル化」(P.9-17)

### RDR フォーマッタと NetFlow エクスポート サポート

- 「RDR フォーマッタ」(P.9-2)
- 「NetFlow」(P.9-2)
- 「データの送信先」(P.9-3)

## RDR フォーマット

RDR フォーマットは、アプリケーションから渡された Raw Data Record (RDR) イベントのストリームの収集、データの外部レポートプロトコル (RdrV1 または NetFlowV9) へのフォーマット、および生成されたレポートの適切な送信先への送信に使用されます。RDR フォーマットによって NetFlow トラフィックのエクスポートが行われる際には、RDR フォーマットのすべての設定が NetFlowV9 レポートのエクスポートに適用されます。RDR タイプに関する詳細とそのフォーマットの説明については、『Cisco Service Control Application for Broadband Reference Guide』を参照してください。

## NetFlow

NetFlow レポートプロトコルは、収集されたレポートデータをその後の収集、集計、保存、および処理のために外部アプリケーションに配信する業界標準のプロトコルです。NetFlow プロトコルオプションは、Service Control ソリューションを幅広い既存データコレクタ/レポートと統合します。

リリース 3.1.0 では、レイヤ 7 アプリケーション エクスポート レポートがサポートされています。

- 「NetFlow の用語」 (P.9-2)
- 「NetFlow エクスポート サポート」 (P.9-3)

## NetFlow の用語

- エクスポート

NetFlow サービスがイネーブルになっており、NetFlowV9 プロトコルを使用した情報のエクスポートを受け持つデバイス (この場合、SCE プラットフォーム内の RDR フォーマット コンポーネント)。

- NetFlow コレクタ

1 つまたは複数のエクスポートからレコードを受け取るデバイス。受け取ったエクスポートパケットを解析して保存することにより、パケットの処理を行います。必要に応じて、ハードディスクに保存する前にレコードを集計できます。

- エクスポート パケット

エクスポートで生成されるパケット。エクスポートのレコードを NetFlow コントローラまで運びます。

- パケット ヘッダー

エクスポートパケットの先頭の部分。パケットヘッダーは、NetFlow バージョン、パケットに含まれるレコードの数、シーケンス番号、観察ドメイン ソース ID などのパケットに関する基本情報を提供します。

- フローセット

よく似た構造を持つフローレコードのコレクションを指す一般的な用語。エクスポートパケットでは、パケットヘッダーの後ろに 1 つまたは複数のフローセットが続きます。フローセットには、次の 2 つのタイプがあります。

- テンプレートフローセット
- データフローセット

- テンプレートフローセット

1 つのエクスポート パケット内で一緒にグループ化されている 1 つまたは複数のテンプレート レコード。

- テンプレート レコード

フロー データ レコード内のフィールドの構造と解釈を定義します。

- データ フローセット

1 つのエクスポート パケット内で一緒にグループ化された 1 つまたは複数のレコード (同一タイプのもの)。各レコードは、その前にテンプレート レコードまたはオプション テンプレート レコードによって定義されているフロー データ レコードかオプション データ レコードのいずれかになります。

- フロー データ レコード

テンプレート レコードに対応するフロー パラメータの値が含まれているデータ レコード。

## NetFlow エクスポート サポート

RDR フォーマッタでは、NetFlowV9 レポートのエクスポートがサポートされており、エクスポート パケットを設定されている送信先に送信することができます。パケットには、テンプレート レコードとデータ レコードが含まれます。テンプレート レコードは、続くデータ レコードのフォーマットを定義します。各エクスポート パケットに、両方のタイプのレコードが含まれている場合もあれば、片方のタイプのレコードしか含まれていない場合もあります。

### NetFlow テンプレート

NetFlowV9 エクスポートにサポートされている各 RDR タイプは、RDR フォーマッタがそれを NetFlow V9 レポートに変換して NetFlow の送信先に送信できるようにする定義済みのマッピングを持ちます。SCE プラットフォームは、複数の RDR タイプのテンプレート レコードを保持しており、それに各 NetFlow データ レコードのその RDR タイプに対応する構造も含まれています。NetFlow テンプレートはすべて定義済みです。ユーザが NetFlow テンプレートを作成したり編集したりすることはできません。

NetFlow エクスポートにサポートされていない RDR タグが NetFlow 送信先に送信されるように設定された場合、このレポートはフォーマットも送信もされず、特殊なカウンタがインクリメントされ、警告が記録されます (show rdr-formatter statistics コマンドの出力の「unsupported-tags」を参照してください)。

テンプレート レコードとデータ レコードを同一のエクスポート パケット内に混在させることはできませんが、テンプレートが関連データ レコードの前になくてもなりません。このため、最初のエクスポート パケットにテンプレートが含まれ、最初にレコードが送信され、設定された間隔 (テンプレート リフレッシュ) で再び送信されて、データ レコードが正しく編成され、正しく読み取られる状態が確保されます。

## データの送信先

- 「カテゴリ」 (P.9-4)
- 「プライオリティ」 (P.9-5)
- 「NetFlow のための DSCP の設定」 (P.9-5)
- 「転送モード」 (P.9-5)
- 「プロトコル」 (P.9-6)
- 「トランスポート タイプ」 (P.9-6)

SCE プラットフォームには、最大 8 の送信先を設定できます（カテゴリにつき 3 つの送信先）。各送信先が、以下のパラメータで定義されます。

- IP アドレス
- ポート番号
- プロトコル（RDRv1 または NetFlow）
- トランスポート タイプ（TCP（RDRv1）または UDP（NetFlow））

送信先には、それが割り当てられている各カテゴリについて、プライオリティが割り当てられます。

図 9-1 に、1 つのカテゴリと 1 つの送信先を持つ最も単純なデータ送信先トポロジを示します。

図 9-1 データ送信先トポロジ : 1 つのカテゴリと 1 つの送信先

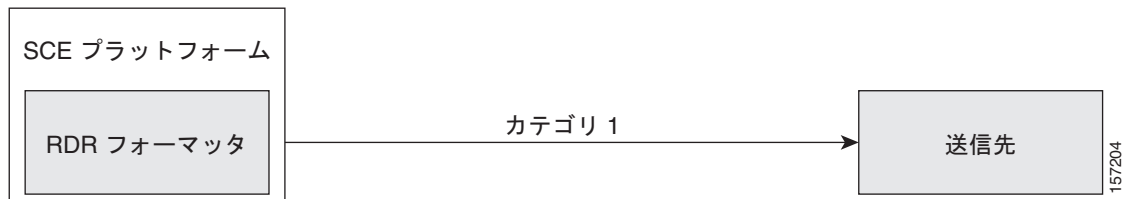
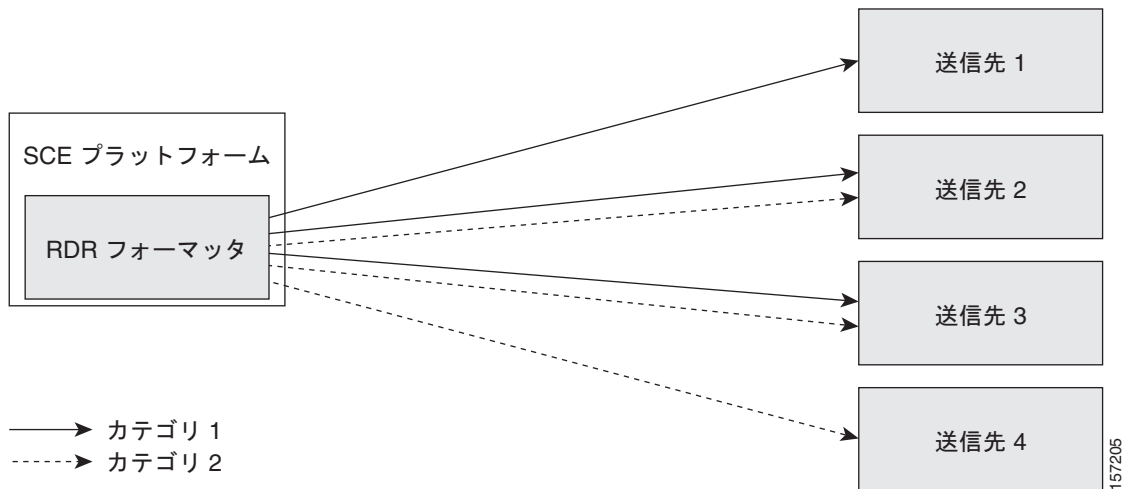


図 9-2 に、2 つのカテゴリと 4 つの送信先を使用する複雑なトポロジを示します。各カテゴリは、4 つの送信先のうち、3 つの送信先にデータを送信できます。

図 9-2 データ送信先トポロジ : 2 つのカテゴリと 4 つの送信先



## カテゴリ

特定のインストレーションでは、タイプに応じて、データを異なるコレクタ サーバに送信する必要があります。たとえば、プリペイド環境では、一部のタイプのデータをプリペイドコレクタに送信して、新しいクォータを取得し、それ以外のデータをメディエーションシステムに送信する必要があります。この場合、データタイプは最大 4 つのグループに分かれます。各グループまたはカテゴリが特定の送信先（複数の場合もある）に割り当てられます。カテゴリは、SCE プラットフォーム上で実行されているアプリケーションによって定義されます。

最大 4 つのカテゴリがサポートされます。このため、送信先は、使用中の各カテゴリに関して設定されている必要があります。各送信先は、複数のカテゴリに割り当てることができます。また、各カテゴリに対して、同一または別のプライオリティを割り当てることができます。

送信先からカテゴリを削除して、必要なカテゴリだけを残すこともできます。すべてのカテゴリが削除されると、送信先も削除されます。

デフォルトでは、カテゴリはカテゴリ 1 ~ 4 で表されます。ただし、ユーザがカテゴリに意味のある名前を定義することも可能です。名前を付けることにより、混乱が低減するので、エラーを防ぐことができます。

## プライオリティ

プライオリティ値は、送信先が所定のカテゴリの送信先に値するかを示すために使用されます。

プライオリティは、冗長転送モードに関連しています。このモードでは、どの接続がプライマリ アクティブ接続であるかを示します。プライオリティ値は、マルチキャスト転送モードでは使用されません。

各送信先には、各カテゴリのプライオリティ値が割り当てられます。明示的に定義していない場合は、設定された最初の送信先に、100（最も高いプライオリティ）のプライオリティがすべてのカテゴリに対して自動的に割り当てられます。

次に、プライオリティ値について覚えておく必要がある重要なポイントの一部を示します。

- 2 つの送信先は、1 つのカテゴリに対して同じプライオリティを持つことができません。カテゴリ内の送信先の各プライオリティ値は、一意のものでないと、意味がありません。
- 1 つのプライオリティ値だけが送信先に割り当てられている場合、プライオリティがその送信先のすべてのカテゴリに自動的に割り当てられます。
- 1 つのカテゴリだけが送信先のプライオリティ値に割り当てられている場合、他のカテゴリの RDR が、指定された送信先に送信されることはありません。
- 指定されたカテゴリのデータをこの送信先に送信するべき場合は、ハイ プライオリティを割り当てます。指定されたカテゴリのデータをこの送信先に送信する必要性が低い場合は、ロー プライオリティを割り当てます。
- 冗長転送モード : ハイ プライオリティをシステム/カテゴリのプライマリ送信先に割り当てます。ロー プライオリティをシステム/カテゴリのセカンダリ送信先に割り当てます。

## NetFlow のための DSCP の設定

NetFlowV9 プロトコルを使用する場合は、NetFlow パケットに割り当てる DSCP 値を設定することにより、プライオリティを定義できます。この DSCP 値は、すべての送信先に対して NetFlow トラフィックの DiffServ レベルを定義します。

## 転送モード

複数の送信先がカテゴリに定義されている場合、システムは、どちらの送信先がデータを受信するかを決める必要があります。これは、転送モードによって決まります。転送モードには、3 つの種類があります。

- **Redundancy** : すべてのレコードがプライマリ（アクティブ）接続だけに送信されます。プライマリ接続が失敗すると、次に高いプライオリティを持つ接続済みの送信先にレコードが送信されます。

- **Multicast** : すべてのレコードがすべての送信先に送信されます。この機能は、データの比率が高い場合には、インスタレーションのパフォーマンスに悪影響を与えることがあります。
- **Simple load balancing** : 一連のレコードをそれぞれ異なる送信先へ、ラウンドロビン方式で送信します。レコードを集約するのは、コレクタの責任です。  
1つの接続が失敗すると、接続されているすべての送信先に履歴バッファのコンテンツが送信されます。



(注)

NetFlow プロトコルを使用している一部のタイプの配置は、マルチキャスト転送モードを必要とします。少なくとも1つのカテゴリに複数の送信先があり、それらの少なくとも1つが NetFlow 送信先である配置では、マルチキャスト転送モードを設定しなければなりません。

## プロトコル

次の2つのプロトコルがサポートされています。

- RDR フォーマッタ : Service Control 独自のプロトコル
- NetFlow V9 : 業界標準のプロトコル

この2つのプロトコルは、同じ配置内で同時に使用できます。ただし、各送信先ではプロトコルを1つずつしかサポートできません。

## トランスポート タイプ

次の2つのトランスポート タイプが使用できます。

- TCP
- UDP

現在のところ、トランスポート タイプは、次のようにして設定されているプロトコルにリンクされません。

- RDRv1 プロトコルには、TCP トランスポート タイプが必要です。
- NetFlow V9 プロトコルには、UDP トランスポート タイプが必要です。

## データの送信先とカテゴリの設定

- 「データ送信先の設定」(P.9-6)
- 「データ カテゴリの設定」(P.9-7)
- 「転送モードの設定」(P.9-11)

## データ送信先の設定

データ送信先の設定に関する CLI コマンドには、次の3つの一般的なカテゴリがあります。

- RDRv1 プロトコルと NetFlow プロトコルの両方に適用される一般的なコマンド
- RDR フォーマッタだけに関連するコマンド (NetFlow エクスポートにも影響する場合があります)
- NetFlowV9 プロトコルと NetFlow エクスポート サポートだけに関連するコマンド

## オプション

RDR と NetFlow エクスポート パケットのいずれでも、データ レコードが SCE プラットフォームから正しい場所に到達するには、以下のパラメータが設定されている必要があります。

- **ip-address** : 送信先の IP アドレス。
- **portnumber** : ポート番号。
- **protocol** : 送信先へのデータの送信に使用されるプロトコル (RDRv1 または NetFlow。プロトコルが割り当てられていない場合、プロトコルは RdrV1 になります)。
- **transport** : トランスポート タイプ。TCP または UDP (このパラメータはプロトコルによって決まるため、省略可能)。

プライオリティ値を割り当てることも可能です。プライオリティは、冗長転送モードでは重要ですが、マルチキャスト モードではそれほど重要ではありません。マルチキャスト モードでは、プライオリティ値が存在すると送信先がレポートを受け取るようになることを知っておいてください。プライオリティとカテゴリの関係については、「[送信先の設定とカテゴリの割り当て](#)」(P.9-8) を参照してください。

---

**ステップ 1** SCE(config)# プロンプトで、**rdr-formatter destination ip-address port portnumber [priority priority] protocol protocol [transport transport]** と入力し、**Enter** キーを押します。

送信先が定義されます。上記の例のように、カテゴリが指定されていない場合は、指定したプライオリティがすべてのカテゴリに指定されます。

---

## データ送信先の設定例

### 例 1

この例は、送信先が 1 つしかなく、NetFlow プロトコルを使用する単純なシステムを設定する方法を示します。送信先が 1 つだけの場合は、プライオリティ値を設定する必要はありません。

```
SCE(config)# rdr-formatter destination 10.1.1.205 port 33000 protocol NetFlowV9 transport udp
```

### 例 2

次に、カテゴリを使用せずに、システムに 2 つの送信先を設定する例を示します。

最初の送信先は、100 のプライオリティが自動的に割り当てられるので、プライオリティを明示的に定義する必要はありません。2 番目の送信先には、プライオリティを明示的に定義する必要があります。

各送信先の両方のカテゴリに同一のプライオリティが自動的に割り当てられますが、カテゴリは無視されるので、問題ありません。

```
SCE(config)# rdr-formatter destination 10.1.1.205 port 33000 protocol RdrV1 transport tcp
SCE(config)# rdr-formatter destination 10.1.1.206 port 33000 priority 80 protocol RdrV1 transport tcp
```

## データ カテゴリの設定

データ カテゴリの定義には次の 2 段階の手順があります。

- カテゴリ名を定義します (任意)。

## ■ データの送信先とカテゴリの設定

- 各カテゴリの正しいプライオリティで送信先を設定し、他のすべての送信先パラメータを設定する方法はいくつか存在しています。また、それらは異なる方法で実行できます。場合によっては事前に計画を立てる必要があります。カテゴリの設定にともなう問題については、次のいくつかの例を参照してください。

**オプション**

次のオプションを使用できます。

- category-number** : カテゴリの数 (1 ~ 4)。
- category-name** : カテゴリに割り当てる名前。

---

**ステップ 1** SCE(config)# プロンプトで、**rdr-formatter category number category-number name category-name** と入力し、**Enter** キーを押します。

指定したカテゴリ番号の名前を定義します。このカテゴリ名は、任意の **rdr-formatter** コマンドでカテゴリ番号の代わりに使用できます。

---

**送信先の設定とカテゴリの割り当て****オプション**

次のオプションを使用できます。

- ip-address** : 送信先の IP アドレス。
- portnumber** : ポート番号。
- category-number** : カテゴリの数 (1 ~ 4)。
- category-name** : カテゴリに割り当てる名前。
- priority** : この送信先の場合にこのカテゴリに割り当てるプライオリティ値 (1 ~ 100)。
- protocol** : 送信先へのデータの送信に使用されるプロトコル (RDRv1 または NetFlow。プロトコルが割り当てられていない場合、プロトコルは RdrV1 になります)。
- transport** : トランスポート タイプ。TCP または UDP (このパラメータはプロトコルによって決まるため、省略可能)。

**一般的なガイドライン**

- 1 つのコマンドで最大 4 つのカテゴリが設定できます。
- カテゴリは、番号でも名前でも定義できます。
- 各カテゴリに異なるプライオリティを割り当てることができます。
- 各カテゴリ内では、各送信先のプライオリティが一意のものである必要があります。

---

**ステップ 1** SCE(config)# プロンプトで、**rdr-formatter destination ip-address port portnumber category [name category-name | number category-number] [priority priority] [category [name category-name | number category-number] [priority priority]] [category [name category-name | number category-number] [priority priority]] [category [name category-name | number category-number] [priority priority]] protocol protocol [transport transport]** と入力し、**Enter** キーを押します。

送信先が定義され、指定したプライオリティ (省略可能) でカテゴリが割り当てられます。

---



## カテゴリを指定したデータ送信先の設定例

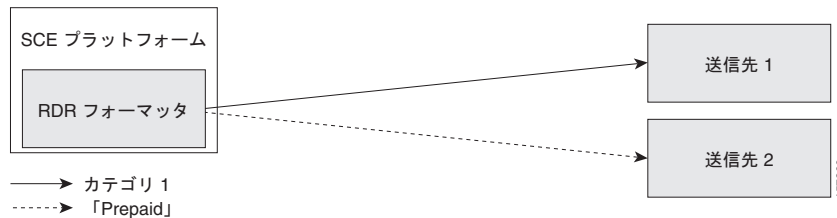
### 例 1

次の図は、1つのカテゴリの名前を定義してから、2つの送信先を設定し、それぞれを別のカテゴリに割り当てます (図 9-3 を参照してください)。

カテゴリ 1 からのデータは、最初の送信先に送信されるので、最初の送信先にそのカテゴリのハイプライオリティが割り当てられており、2番目の送信先にはプライオリティが割り当てられていません。

カテゴリ 2 (プリペイド) からのすべてのデータが 2 番目の送信先に送信されるので、カテゴリ 2 に割り当てられているプライオリティは、2 番目の送信先だけに割り当てられており、最初の送信先には割り当てられていません。

図 9-3 送信先の設定 : 2つのカテゴリと 2つの送信先



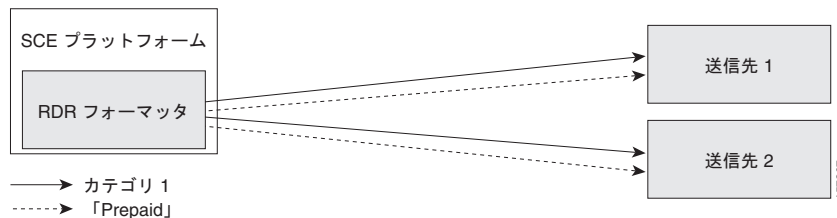
どちらかの送信先への接続損失が生じると、接続が再確立されるまで該当するカテゴリのデータの伝送が中断されます。両方のカテゴリに定義されている冗長接続はありません。

```
SCE(config)# rdr-formatter category number 2 name prepaid
SCE(config)# rdr-formatter destination 10.1.1.205 port 33000 category number 1 priority 90
protocol RdrV1 transport tcp
SCE(config)# rdr-formatter destination 10.1.1.206 port 33000 category name prepaid
protocol RdrV1 transport tcp
```

### 例 2

この例は上記の例と類似していますが、この例では、プライオリティを割り当てないのではなく、各送信先の第 2 のカテゴリにロープライオリティが割り当てられています (図 9-4 を参照してください)。これにより、接続のいずれかで問題が発生した場合に、各送信先が他方のバックアップとして機能できます (冗長転送モード)。

図 9-4 送信先の設定 : 冗長モードでの 2つのカテゴリ



```
SCE(config)# rdr-formatter category number 2 name prepaid
SCE(config)# rdr-formatter destination 10.1.1.205 port 33000 category name prepaid
priority 90 category number 1 priority 25 protocol RdrV1 transport tcp
SCE(config)# rdr-formatter destination 10.1.1.206 port 33000 category number 1 priority 80
category name prepaid priority 20 protocol RdrV1 transport tcp
```

**例 3**

この例では、1つのカテゴリを最初の送信先だけに割り当てて、もう一方のカテゴリが2番目の送信先をプライマリ送信先として使用し、最初の送信先をセカンダリ送信先として使用する2つの方法を示します。

```
SCE(config)# rdr-formatter category number 2 name prepaid
SCE(config)# rdr-formatter destination 10.1.1.205 port 33000 category name prepaid
priority 90 category number 1 priority 10 protocol RdrV1 transport tcp
SCE(config)# rdr-formatter destination 10.1.1.206 port 33000 category number 1 priority 95
protocol RdrV1 transport tcp
```

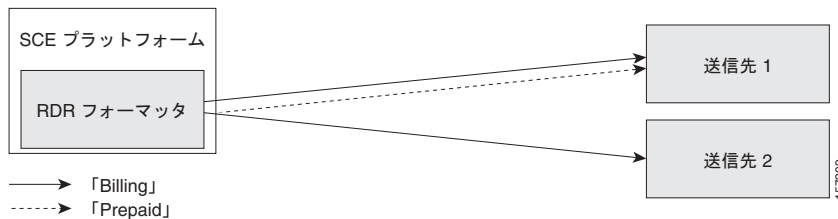
次の例では、すべてのプライオリティ値がかなり高いように見えます。ただし、これは、プライマリ送信先となる送信先を決めるカテゴリに対するプライオリティの相対値です。

```
SCE(config)# rdr-formatter category number 2 name prepaid
SCE(config)# rdr-formatter destination 10.1.1.205 port 33000 priority 90 protocol RdrV1
transport tcp
SCE(config)# rdr-formatter destination 10.1.1.206 port 33000 priority 95 protocol RdrV1
transport tcp
SCE(config)# no rdr-formatter destination 10.1.1.206 port 33000 category name prepaid
protocol RdrV1 transport tcp
```

**例 4**

この例は、1つのカテゴリ（プリペイド）が1つの送信先に割り当てられており、他のカテゴリ（課金）はマルチキャストモードで両方の送信先に送信される、より複雑な設定を示します（[図 9-5](#)を参照してください）。

**図 9-5 送信先の設定 : 2つのカテゴリと2つのモード**



転送モードは、1つのカテゴリだけでなく、RDR フォーマッタ全体に定義されます。「プリペイド」カテゴリは1つの送信先だけに送信されるので、転送モードは関係ありません。一方、「課金」カテゴリは2つの異なる送信先に送信されるため、転送モードに関連します。

```
SCE(config)# rdr-formatter forwarding-mode multicast
SCE(config)# rdr-formatter category number 1 name billing
SCE(config)# rdr-formatter category number 2 name prepaid
SCE(config)# rdr-formatter destination 10.1.1.205 port 33000 priority 40 protocol
NetFlowV9 transport udp
SCE(config)# no rdr-formatter destination 10.1.1.205 port 33000 category name prepaid
protocol NetFlowV9 transport udp
SCE(config)# rdr-formatter destination 10.10.10.96 port 33000 category name billing
priority 90 protocol NetFlowV9 transport udp
SCE(config)# rdr-formatter destination 10.1.96.0 port 33000 category name prepaid priority
80prepaid priority 80 protocol NetFlowV9 transport udp
```

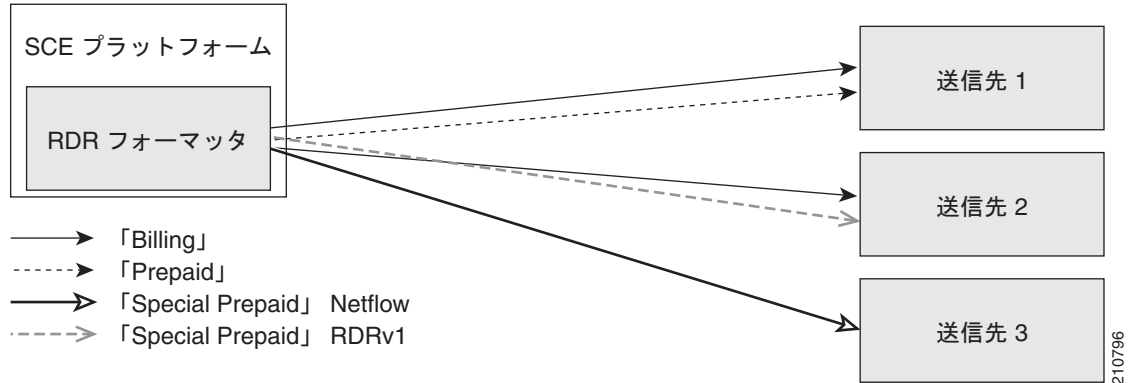
**例 5**

最後に、次の例は、3つのカテゴリを持ち、3つの送信先を持つ設定を示します（[図 9-6](#)を参照してください）。

- カテゴリ 1 : 「課金」、RDRv1 プロトコル、送信先 1 へ送信
- カテゴリ 2 : 「プリペイド」、RDRv1 プロトコル、送信先 1 と送信先 2 へ送信

- カテゴリ 3 : 「特別プリペイド」、NetFlow V9 プロトコル、送信先 3 へ送信、RDRv1 プロトコルが送信先 2 へ送信

図 9-6 送信先の設定 : 3 つのカテゴリと 2 つのプロトコル



```
SCE(config)# rdr-formatter forwarding-mode multicast
SCE(config)# rdr-formatter category number 1 name billing
SCE(config)# rdr-formatter category number 2 name prepaid
SCE(config)# rdr-formatter category number 3 name special-prepaid
SCE(config)# rdr-formatter destination 10.1.1.205 port 33000 category name billing
priority 90 category name prepaid priority 80 protocol RdrV1 transport tcp
SCE(config)# rdr-formatter destination 10.10.10.96 port 33000 category name prepaid
priority 90 category name special-prepaid priority 80 protocol RdrV1 transport tcp
SCE(config)# rdr-formatter destination 10.1.1.206 port 33000 category name special-prepaid
priority 90 protocol NetFlowV9 transport udp
```

## 転送モードの設定

少なくとも 1 つのカテゴリに複数の送信先があり、それらの少なくとも 1 つが NetFlow 送信先である配置では、マルチキャスト転送モードを設定しなければなりません。

## オプション

次の転送モードが指定できます。

- **redundancy** : すべてのレコードがプライマリ（アクティブ）接続だけに送信されます。プライマリ接続が失敗すると、次に高いプライオリティを持つ接続済みの送信先にレコードが送信されます。
- **multicast** : すべてのレコードがすべての送信先に送信されます。この機能は、データの比率が高い場合には、インスタレーションのパフォーマンスに悪影響を与えることがあります。
- **load-balancing** : 一連のレコードをそれぞれ異なる送信先へ、ラウンドロビン方式で送信します。レコードを集約するのは、コレクタの責任です。

**ステップ 1** SCE(config)# プロンプトで、**rdr-formatter forwarding-mode mode** と入力し、**Enter** キーを押します。

指定した転送モードが設定されます。

## 転送モードの設定例

次に、転送モードを `multicast` に設定する例を示します。

```
SCE(config)# rdr-formatter forwarding-mode multicast
```

# RDR フォーマッタの設定

- 「オプション」 (P.9-12)
- 「RDR フォーマッタをイネーブルにする方法」 (P.9-12)
- 「RDR フォーマッタをディセーブルにする方法」 (P.9-12)
- 「RDR フォーマッタの履歴バッファのサイズを設定する方法」 (P.9-13)

## オプション

次のオプションは、RDR フォーマッタだけに関連するものです。

- RDR フォーマッタのイネーブル化とディセーブル化。
- RDR フォーマッタの履歴バッファのサイズの設定。ただし、履歴バッファのサイズは 0 バイト（デフォルト値）でなければなりません。その他の値は、RDR の複製を発生させる可能性があります。
- RDR のカテゴリへのダイナミック マッピング（「RDR からカテゴリへのダイナミック マッピングの設定」 (P.9-14) を参照してください）。

## RDR フォーマッタをイネーブルにする方法

- 
- ステップ 1** SCE(config)# プロンプトで、**service rdr-formatter** と入力し、**Enter** キーを押します。  
RDR フォーマッタがイネーブルになります。
- 

## RDR フォーマッタをディセーブルにする方法

- 
- ステップ 1** SCE(config)# プロンプトで、**no service rdr-formatter** と入力し、**Enter** キーを押します。  
RDR フォーマッタがディセーブルになります。
-

## RDR フォーマッタの履歴バッファのサイズを設定する方法

バッファ サイズは 0 に設定しなければなりません。

- 
- ステップ 1** SCE(config)# プロンプトで、**rdr-formatter history-size 0** と入力し、Enter キーを押します。  
RDR フォーマッタの履歴バッファのサイズが設定されます。
- 

## NetFlow エクスポート サポートの設定

- 「オプション」(P.9-13)
- 「NetFlow の DSCP 値を設定する方法」(P.9-13)
- 「テンプレート リフレッシュ間隔を設定する方法」(P.9-14)

### オプション

次のオプションは、NetFlow エクスポート レポート (RDR フォーマッタ内) にだけ関連するものです。

- プライオリティ設定のための指定した送信先への NetFlow エクスポート パケットに対する DSCP 値の割り当て。  
DSCP 値は、0 ~ 63 の範囲内でなくてはならず、16 進形式で入力します。
- テンプレート レコードをエクスポートする頻度の設定 (テンプレート リフレッシュ間隔)。

## NetFlow の DSCP 値を設定する方法

### オプション

次のオプションを使用できます。

- **dscp-value** : すべての送信先に対する NetFlow パケットに割り当てられる DSCP 値 (16 進形式での 0 ~ 63 の値)

- 
- ステップ 1** SCE(config)# プロンプトで、**rdr-formatter protocol NetFlowV9 dscp dscp-value** と入力し、Enter キーを押します。

NetFlow エクスポート サポートの DSCP 値が設定されます。

---

## テンプレート リフレッシュ間隔を設定する方法

### オプション

次のオプションを使用できます。

- **ip-address** : 送信先 IP アドレス
- **port-number** : 送信先ポート番号
- **timeout-value** : テンプレート レコードをエクスポートする頻度 (秒単位、1 ~ 86400 の値)

---

**ステップ 1** SCE(config)# プロンプトで、**rdr-formatter destination ip-address port port-number protocol NetFlowV9 template data timeout timeout-value** と入力し、**Enter** キーを押します。

テンプレート リフレッシュ間隔が設定されます。

---

## RDR からカテゴリへのダイナミック マッピングの設定

RDR から複数のカテゴリへのダイナミック設定がサポートされています。

各 RDR タグが、カテゴリのリストを持ちます。デフォルト カテゴリは、アプリケーションがロードされたときに割り当てられたカテゴリです。

RDR タグに対するカテゴリの設定は、マッピングの追加および削除によって行われます。ユーザが、RDR タグのカテゴリへのマッピングを追加したり、マッピングを削除したりできます。デフォルト マッピングも削除できます。タグからすべてのカテゴリを削除すると、マッピングされたカテゴリがない限り、そのタグは無視されます。

ユーザは、カテゴリを追加したり削除したりするには、RDR タグ ID とカテゴリ番号を指定しなければなりません。設定は、アプリケーション コンフィギュレーションの一部として保存されます。

### マッピングの設定

マッピングの追加と削除には、ここで説明するコマンドを使用します。

- 「オプション」(P.9-14)
- 「カテゴリへのマッピングを追加する方法」(P.9-15)
- 「カテゴリからマッピングを削除する方法」(P.9-15)
- 「指定した RDR タグのデフォルト マッピングを復元する方法」(P.9-15)

### オプション

次のオプションを使用できます。

- **tag-umber** : 16 進数として与えられる完全な 32 ビット値。すでにアプリケーションによってフォーマッタ内で RDR タグが設定されている必要があります。
- **category-number** : RDR タグのマッピング先となるカテゴリの番号 (1 ~ 4)。

## カテゴリへのマッピングを追加する方法

**ステップ 1** SCE(config)# プロンプトで、**rdr-formatter rdr-mapping (tag-id tag-number category-number category-number)** と入力し、**Enter** キーを押します。

テーブルにすでに同じタグとカテゴリ番号でのマッピングが含まれていた場合は、エラーが発行され、何も行われません。

## カテゴリからマッピングを削除する方法

**ステップ 1** SCE(config)# プロンプトで、**no rdr-formatter rdr-mapping (tag-id tag-number category-number category-number)** と入力し、**Enter** キーを押します。

## 指定した RDR タグのデフォルト マッピングを復元する方法

**ステップ 1** SCE(config)# プロンプトで、**default rdr-formatter rdr-mapping tag-id tag-number** と入力し、**Enter** キーを押します。

## データ送信先の設定と統計情報の表示

- 「現在の RDR フォーマッタ設定を表示する方法」 (P.9-16)
- 「現在の RDR フォーマッタの統計情報を表示する方法」 (P.9-16)

次のコマンドを使用して、RDR フォーマッタの設定と統計情報を表示できます。

- **show rdr-formatter**
- **show rdr-formatter connection-status**
- **show rdr-formatter counters**
- **show rdr-formatter destination**
- **show rdr-formatter enabled**
- **show rdr-formatter forwarding-mode**
- **show rdr-formatter rdr-mapping**
- **show rdr-formatter statistics**
- **show rdr-formatter protocol NetFlowV9 dscp**

その他の **show rdr-formatter** コマンドの詳細については、『[Cisco SCE8000 CLI Command Reference](#)』を参照してください。

## 現在の RDR フォーマッタ設定を表示する方法

データ送信先のすべての設定を表示することもできれば、特定のパラメータだけを表示することもできます。

- ステップ 1** SCE> プロンプトで、**show rdr-formatter** と入力し、**Enter** キーを押します。  
現在の RDR フォーマッタの設定が表示されます。

### RDR フォーマッタの設定の表示例

次に、現在の RDR フォーマッタの設定を表示する例を示します。

```
SCE#show rdr-formatter
Status: enabled
Connection is: up
Forwarding mode: redundancy
Connection table:
-----|
Collector | Port|Status|          Priority per Category:          |
IP Address / | | |-----|
Host-Name | | |Category1 |Category2 |Category3 |Category4 |
-----|
10.1.1.205 |33000|Up |100 primary|100 primary|100 primary|100 primary|
10.1.1.206 |33000|Down |60 |60 |60 |60 |
10.12.12.12 |33000|Up |40 |40 |40 |40 |
-----|
RDR:   queued:    0 , sent:4460807,  thrown:    0,  format-mismatch:0
UM:    queued:    0 , sent:      0,  thrown:    0
Logger: queued:    0 , sent:     39,  thrown:    0
Errors: thrown:    0
Last time these counters were cleared: 20:23:05 IST WED March 14 2007
```

## 現在の RDR フォーマッタの統計情報を表示する方法

- ステップ 1** SCE> プロンプトで、**show rdr-formatter statistics** と入力し、**Enter** キーを押します。  
現在の RDR フォーマッタの統計情報が表示されます。

### 現在の RDR フォーマッタ統計情報の表示例

次の例は、RDRv1 プロトコルと NetFlow プロトコルの両方を使用している配置での現在の統計情報を表示する方法を示します。

```
SCE#show rdr-formatter statistics
RDR-formatter statistics:
=====
Category 1:
sent:                1794517
in-queue:            0
thrown:              0
format-mismatch:     0
unsupported-tags:    1701243
```



```
rate:                2 RDRs per second
max-rate:            64 RDRs per second
Category 2:
sent:                12040436
in-queue:           0
thrown:              0
format-mismatch:    0
unsupported-tags:    0
rate:                12 RDRs per second
max-rate:            453 RDRs per second
Category 3:
sent:                0
in-queue:           0
thrown:              0
format-mismatch:    0
unsupported-tags:    0
rate:                0 RDRs per second
max-rate:            0 RDRs per second
Category 4:
sent:                0
in-queue:           0
thrown:              0
format-mismatch:    0
unsupported-tags:    0
rate:                0 RDRs per second
max-rate:            0 RDRs per second
Destination:         10.56.201.50 Port: 33000 Status: up
Sent:                13835366
Rate:                211 Max: 679
Last connection establishment: 17 hours, 5 minutes, 14 seconds
Destination:         10.56.204.7 Port: 33000 Status: up
Sent:                12134054
Rate:                183 Max: 595
Sent Templates:     13732
Sent Data Records:  12134054
Refresh Timeout (Sec): 5
Last connection establishment: 17 hours, 5 minutes, 15 seconds
```

## ラインカードによる RDR 送信のディセーブル化

**silent** コマンドは、ラインカードによるデータレコードの送信をディセーブルにします。RDR と NetFlow 両方のエクスポートパケットの送信が停止されます。

ラインカードによるレポート送信を有効にするには、このコマンドの **no** 形式を使用します。

---

**ステップ 1** SCE(config if)# プロンプトで、**silent** と入力し、**Enter** キーを押します。

---

ラインカードによるデータレコード生成をイネーブルにするには、ここで説明するコマンドを使用します。

---

**ステップ 1** SCE(config if)# プロンプトで、**no silent** と入力し、**Enter** キーを押します。

---

## RDR 集約をディセーブルにする

大規模な配置で、各トラフィック プロセッサが個別に独自のレコードを CM に送信する場合は、CM に到達する RDR の数が膨大になります。そのため、SCE プラットフォームでは特定の RDR が集約され、送信される情報の使用可能性に影響を与えることなく、CM への負荷が削減されます。基本的に、コントロール プロセッサはすべてのトラフィック プロセッサからレコードを受信しますが、送信するのは各レポート期間につき、すべての CPU のデータを集約した 1 つのレコードのみです。

RDR の集約機能は、グローバルなレコードにのみ関係します。具体的には、定期的なレコードのみが集約されます。これは、他のレコードは単一のトランザクションやフローなどのイベントに関連するため、プロセッサ間で集約することができないからです。これらを集約すると、必要な精度が失われてしまいます。

現時点では、次の RDR が集約されます。

- 仮想リンク使用 RDR (VLUR)
- リンク使用 RDR (LUR)
- パッケージ使用 RDR (PUR)

RDR 集約は、特定の RDR タイプについて、または集約されたすべての RDR についてディセーブルにすることができます。

---

**ステップ 1** SCE(config if)# プロンプトで、**no periodic-records aggregate-by-cpu [all | LUR | PUR | VLUR]** と入力し、**Enter** キーを押します。

---

RDR 集約を再度イネーブルにするには、次のコマンドを使用します。

---

**ステップ 1** SCE(config if)# プロンプトで、**periodic-records aggregate-by-cpu [all | LUR | PUR | VLUR]** と入力し、**Enter** キーを押します。

---

現在の RDR 集約を表示するには、次のコマンドを使用します。

---

**ステップ 1** SCE> プロンプトに、**show interface LineCard 0 periodic-records aggregation** を入力して、**Enter** キーを押します。

---