



SAN Telemetry Streaming の構成

この章では、SAN Telemetry Streaming 機能とその構成方法について説明します。

- [SAN Telemetry Streaming の設定の機能履歴](#) (1 ページ)
- [SAN Telemetry Streaming の概要](#) (2 ページ)
- [SAN Telemetry Streaming の注意事項と制約事項](#) (5 ページ)
- [gRPC エラーの動作](#) (6 ページ)
- [SAN テレメトリ ストリーミングのエンコーディング](#) (7 ページ)
- [SAN テレメトリ ストリーミングの設定](#) (8 ページ)
- [例 : SAN テレメトリ ストリーミングの設定](#) (11 ページ)
- [SAN テレメトリ ストリーミングの設定と統計情報の表示](#) (14 ページ)
- [SAN テレメトリ ストリーミングのトラブルシューティング](#) (20 ページ)

SAN Telemetry Streaming の設定の機能履歴

表 1: SAN Telemetry Streaming の設定の機能履歴

機能名	リリース	機能情報
トランシーバーパラメータのストリーミング	9.2(2)	FC トランシーバーパラメータストリーミングのサポートが追加されました。
SAN Telemetry Streaming	8.4(1)	NVMe フローメトリックを使用して、 <i>fabric_telemetry.proto</i> ファイルを更新しました。
SAN Telemetry Streaming	8.3(2)	コンパクト Google Protocol Buffers (GPB) エンコーディングのサポートが追加されています。

機能名	リリース	機能情報
SAN Telemetry Streaming	8.3(1)	<p>ストリーミング アナリティクスやインターフェイスの統計情報の機能を DCNM などの受信者に提供します。</p> <p>次に、追加されたコマンドを示します。</p> <ul style="list-style-type: none"> • certificate <i>certificate_path host_name</i> • destination-group <i>id</i> • destination-profile • dst-grp <i>id</i> • feature telemetry • {ip ipv6} address <i>address</i> port <i>number</i> [protocol <i>procedural-protocol</i> encoding <i>encoding-protocol</i>] • path <i>sensor_path</i> • sensor-group <i>id</i> • show run telemetry • show telemetry { control { database [destination-groups destinations sensor-groups sensor-paths subscriptions] stats } data collector { brief details } pipeline stats transport <i>session_id</i> [errors stats] } • snsr-grp <i>id</i> sample-interval <i>interval</i> • subscription <i>id</i> • telemetry • use-retry size <i>buffer_size</i>
インターフェイスの統計情報	8.3(1)	<p>ファイバ チャネル インターフェイスからトラフィックおよびエラー カウンタのデータをストリーミングできます。</p>

SAN Telemetry Streaming の概要

Cisco NX-OS には、ネットワークからデータを収集するための複数のメカニズム（Simple Network Management Protocol (SNMP)、CLI、Syslog など）があります。SAN Telemetry Streaming 機能は、DCNM などの 1 つ以上のアップストリーム レシーバに、分析する特定のデータをストリーミングするために使用されます。SAN Analytics で使用されるプル モデルは、クライアントから要求された場合にのみサーバーからデータを送信するために使用されます。

一般的には、クライアントに継続的にデータをストリーミングするために使用されるプッシュ（フェッチ）モデルを使用してスイッチからデータが収集されます。SAN Telemetry Streaming はプッシュ モデルを有効にし、モニタリング データにほぼリアルタイムでアクセスできるようにします。

収集されたトラフィックおよびエラーカウンタのデータは、SAN Telemetry Streaming の構成でセンサー グループにセンサー パスを追加することで、DCNM、サードパーティ製デバイス、またはアプリケーションにストリーミングできます。詳細については、[SAN テレメトリ ストリーミングの設定（8 ページ）](#) を参照してください。



(注) Cisco MDS NX-OS リリース 8.3(1) では、テレメトリ ペイロードに追加されたバージョン番号は 1.0.0.1 です。

インターフェイスの統計情報のストリーミング

インターフェイスの統計情報のストリーミングを使用すると、ファイバチャネルインターフェイスからトラフィックおよびエラーカウンタのデータをストリーミングできます。トラフィックおよびエラーカウンタの収集はデフォルトで有効になっているため、設定または無効化することはできません。65 を超えるインターフェイスの統計情報カウンタを使用できます。インターフェイスの統計情報をサポートしているモジュールの詳細については、[SAN アナリティクス のハードウェア要件](#) を参照してください。

サポートされているインターフェイス カウンタの一覧については、[インターフェイス カウンタ](#) を参照してください。

トランシーバパラメータ ストリーミング

トランシーバパラメータ ストリーミングは、トランシーバに関する情報を定期的に収集し、それをレシーバにストリーミングします。この情報は、動作中の DOM (Diagnostic Optical Monitoring) データと、監視対象の各トランシーバのベンダー名、モデル番号、シリアル番号、およびスイッチのタイムスタンプに関する静的データの両方で構成されます。これにより、ローカル NX-OS オンスイッチ トランシーバパラメータのしきい値モニタリングを介した集中型および拡張トランシーバ モニタリングが可能になります。

トランシーバ DOM の動作パラメータを経時的に分析することで、トランシーバのパフォーマンスの問題を特定できます。たとえば、ビット エラーやフレーム CRC などのインターフェイスエラーをトランシーバの受信電力レベルと関連付けることで、ケーブルの断続的な問題を特定することができます。タイムスタンプは、時間のシーケンシングや、他のデータやログとの関連付けに使用できます。

トランシーバパラメータストリーミングセンサーは、ローカルスイッチ トランシーバ データのみを収集するか、ローカルおよびピア トランシーバ データの両方を収集するように定義できます。



(注) ピア トランシーバ データを監視するには、ピア デバイスがインバンド FC 読み取り診断パラメータ (RDP) ELS 要求をサポートしている必要があります。

この機能は次のコンポーネントで構成されています。

- スイッチでの収集：表 2: ストリーミング トランシーバパラメータ (4 ページ) にリストされているトランシーバのパラメータは定期的に収集されます。これらは、トランシーバパラメータ ストリーミングとは関係なく、スイッチ上の NX-OS によってローカルに監視されます。
- レシーバへのストリーミング：テレメトリ構成コマンドを使用して、ストリーミングするインターフェイスの範囲と、トランシーバパラメータのストリーミング インターバルを指定します。ストリーミングは、古いデータがストリーミングされるのを避けるために、トランシーバが動作してから 10 分後に開始されます。次に、レシーバはデータを監視および分析することができます。



(注) トランシーバパラメータのストリーミングは、ファイバチャネルポートでのみサポートされます。

表 2: ストリーミング トランシーバパラメータ (4 ページ) は、ストリーミングされるトランシーバパラメータのリストを表示します。

表 2: ストリーミング トランシーバパラメータ

トランシーバパラメータ	ユニット
温度	摂氏 (C)
電圧	ボルト (V)
電流	ミリアンペア (mA)
Tx Power	デシベル ミリワット (dBm)
Rx 電力	デシベル ミリワット (dBm)
ベンダー名 (Vendor Name)	—
モデル番号	—
シリアル番号	—
スイッチ タイムスタンプ	—

SAN Telemetry Streaming の注意事項と制約事項

- **feature telemetry** コマンドが有効になっている場合は、Cisco MDS NX-OS リリース 8.3(1) よりも前のリリースにダウングレードする前に、**no feature telemetry** コマンドを使用してこの機能を無効にします。
- Cisco MDS NX-OS リリース 8.3(2) の前は、SAN Telemetry Streaming は Google リモートプロシージャコール (gRPC) 転送を介した Google Protocol Buffers (GPB) エンコーディングのみサポートしていました。Cisco MDS NX-OS リリース 8.3(2) からは、コンパクト GPB エンコーディングのサポートが追加されています。宛先グループの下にあるすべての宛先とサブスクリプションの下にあるすべての宛先グループが同じエンコーディングタイプであることを確認します。



(注) GPB キー値エンコーディングは、単純に GPB として参照されています。GPB は、configuration および show コマンドで GPB キー値の代わりに使用されています。

- Cisco DCNM SAN Insights を使用している場合は、Cisco DCNM SAN Insights で SAN Telemetry Streaming 機能を構成できます。スイッチでこの機能を構成する必要はありません。詳細については、[Cisco DCNM SAN 管理コンフィギュレーションガイド \[英語\]](#) の「Configuring SAN Insights」セクションを参照してください。
- ストリーミング サンプル間隔 (**snsr-grp id sample-interval interval**)、ポート サンプリング間隔 (**analytics port-sampling module number size number interval seconds**)、およびプッシュクエリ間隔 (**analytics query “query_string” name query_name type periodic [interval seconds] [clear] [differential]**) は、同じ値に設定することをお勧めします。また、最初にプッシュクエリ間隔、次にポート サンプリング間隔、最後にストリーミング サンプル間隔を変更または設定することをお勧めします。
- サポートされている最小のストリーミング サンプル間隔は 30 秒です。プッシュクエリ間隔、ポート サンプリング間隔、およびストリーミング サンプル間隔は、最小推奨値の 30 秒以上にし、同じ値に設定することをお勧めします。最小値未満の間隔を設定すると、望ましくないシステム動作が発生する可能性があります。
- インターフェイスの統計情報のストリーミングは、Cisco NPV モードで動作している Cisco MDS 9132T スイッチではサポートされていません。
- 最大 2 つの管理レシーバ (宛先) がサポートされます。ただし、最適なパフォーマンスを実現するために、レシーバは 1 つだけ設定することをお勧めします。
- 複数のレシーバ (Cisco DCNM またはサードパーティ製デバイスやアプリ) を構成する場合は、同じ宛先グループの下に構成することをお勧めします。複数の Cisco DCNM レシーバがある場合は、それらのレシーバを手動で同じ宛先グループに設定する必要があります。

- SAN Telemetry Streaming レシーバが機能を停止すると、他のレシーバでデータフローが中断されます。失敗したレシーバを再起動します。レシーバの再起動方法については、レシーバのマニュアルを参照してください。

レシーバが遅延なく稼働していて、管理ポートでパケットドロップが発生していない場合、テレメトリ データ ストリーミングは均一です。レシーバまたはネットワークの遅さが原因で gRPC 転送に遅延が発生している場合、データ収集が中断され、システムメモリの制限によりスイッチ側でデータがドロップされる可能性があります。この問題が発生するかどうかは、ストリーム出力する ITL 数の程度、およびネットワークの遅延または速度低下の程度によって決まります。 **show telemetry control database sensor-groups**、**show telemetry transport session_id errors**、およびテレメトリ **syslog** コマンドを使用して、センサー グループ レベルでのドロップ数とすべての転送遅延がある場合はその転送ステータスを確認します。詳細については、[SAN テレメトリ ストリーミングのトラブルシューティング \(20 ページ\)](#) を参照してください。



- (注) ネットワークの速度低下が解決しない場合、または分析データの転送やストリーミング速度を 25 時間以上遅延させている継続的なネットワークの低下が発生している場合、転送セッションが永久的に無効化されて Syslog メッセージが生成されます。この問題が解決した後は、対応する宛先グループの下の IP アドレスを削除して構成することでストリーミングを再開できます。設定の詳細については、「[SAN テレメトリ ストリーミングの設定 \(8 ページ\)](#)」を参照してください。

gRPC エラーの動作

gRPC レシーバが 20 個のエラー (gRPC エラーの 1 つまたは両方) をスイッチに送信した後、スイッチクライアントは gRPC レシーバへの接続を無効にします。レシーバからの応答の受信に 30 秒以上かかり、この状態が 25 時間継続している場合、各転送セッションは「無効」とマークされます。gRPC レシーバ有効にするには、宛先グループの下の宛先 IP アドレスの設定を解除して再設定する必要があります。生成されたエラーを表示するには、**show telemetry transport session_id errors** コマンドを使用します。設定の詳細については、[SAN テレメトリ ストリーミングの設定 \(8 ページ\)](#) を参照、エラーについては、[SAN テレメトリ ストリーミングのトラブルシューティング \(20 ページ\)](#) を参照してください。

以下は gRPC エラーです。

- gRPC クライアントがセキュアな接続に対して誤った証明書を送信する。
- gRPC レシーバでのクライアントメッセージの処理に時間がかかりすぎて、タイムアウトが発生する。別のメッセージ処理スレッドを使用してメッセージを処理することで、タイムアウトを回避している。

SAN テレメトリ ストリーミングのエンコーディング

SAN テレメトリ ストリーミングでは、次のエンコーディングが使用されます。

- **GPB キー値** : Cisco MDS NX-OS リリース 8.3(2) の前は、GPB キー値が唯一サポートされているエンコーディングでした。このエンコーディングで使用されるキーは文字列で、自己記述型です。ただし、このエンコーディングで使用されるデータのサイズは、コンパクト GPB エンコーディングよりも大きくなります。このタイプのエンコーディングでは、中間プロセスなしでデータを簡単に分析できます。[key フィールドの詳細については、フローメトリック](#)を参照してください。
- **コンパクト GPB** : Cisco MDS NX-OS リリース 8.3(2) からは、コンパクト GPB エンコーディングのサポートが追加されています。このエンコーディングで使用されるキーは整数です。そのため、このエンコーディングで使用されるデータのサイズはGPB-KVエンコーディングよりも小さくなります。ただし、整数をそれぞれのメトリックに復号化するには復号化テーブルが必要です。コンパクト GPB の復号化テーブルは `.proto` ファイルです。コンパクト GPB を使用する場合、すべての `path analytics: query_name` クエリに対して `telemetry_bis.proto` ファイルを使用し、データ ストリームを解析するためにコレクタにこのファイルをアップロードする必要があります。



- (注) インターフェイスの統計情報のストリーミング (`path show_stats`) の場合は、GPB-KV エンコーディングのみサポートされます。

次に、コンパクト GPB `.proto` ファイルで使用されるテレメトリ フィールドのスニペットの表示例を示します。

```
message Telemetry {
  ...
  repeated TelemetryField data_gpbkv = 11;
  TelemetryGPBTable data_gpb = 12;
  ...
  message TelemetryGPBTable {
    repeated TelemetryRowGPB row = 1;
  }
  message TelemetryRowGPB {
    uint64 timestamp = 1;
    bytes keys = 10;
    bytes content = 11;
  }
}
```

この例では、コンパクト GPB の `.proto` ファイルで使用されているフィールドは、`data_gpb` フィールドの下に含まれています。TelemetryRowGPB メッセージ構造の `key` フィールドは、`.proto` ファイル名 (`fabric_telemetry`) を伝送し、`content` フィールドは `.proto` ファイルからのフィールドを伝送します。

コンパクト GPB で使用される `.proto` ファイルの詳細については、[SAN テレメトリ ストリーミング Proto ファイル](#)を参照してください。

SAN テレメトリ ストリーミングの設定



- (注) Cisco DCNM SAN Insights を使用している場合は、Cisco DCNM SAN Insights で SAN Telemetry Streaming 機能を構成できます。スイッチでこの機能を構成する必要はありません。詳細については、[Cisco DCNM SAN 管理コンフィギュレーションガイド \[英語\]](#) の「Configuring SAN Insights」セクションを参照してください。

後続の図は、センサーと宛先グループのさまざまな設定方法を示しています。

図 1: 同じ宛先グループにマッピングされているセンサーグループ

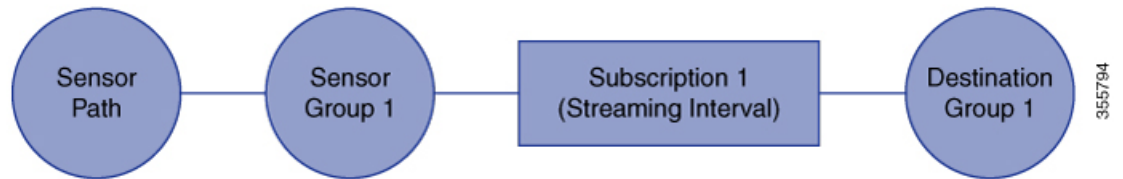


図 2: 別の宛先グループにマッピングされているセンサーグループ

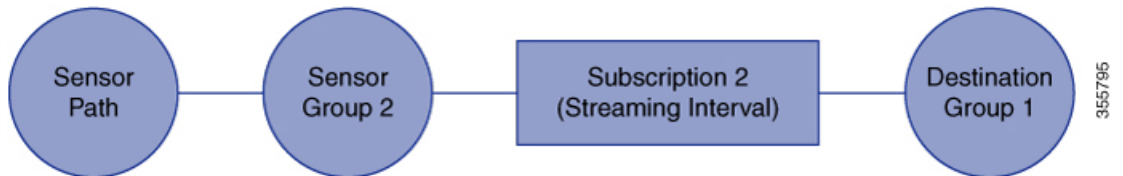


図 3: 複数の宛先グループにマッピングされている 1つのセンサーグループ

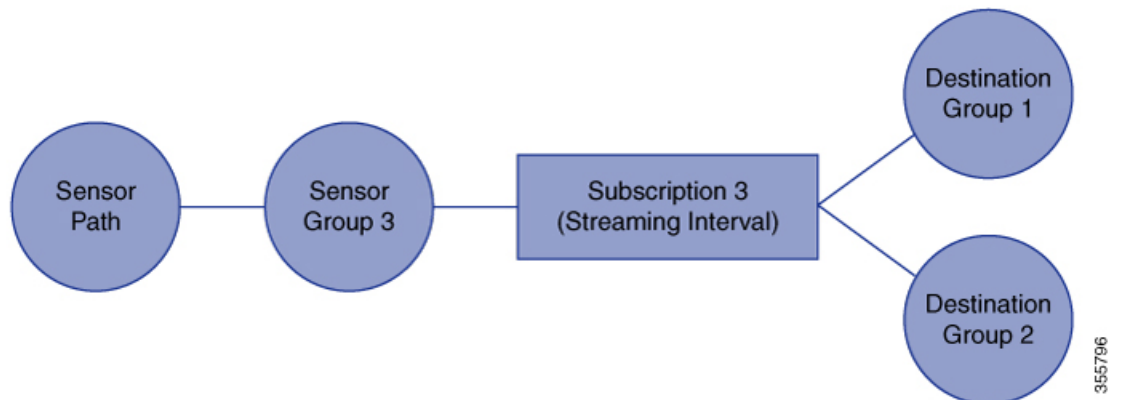
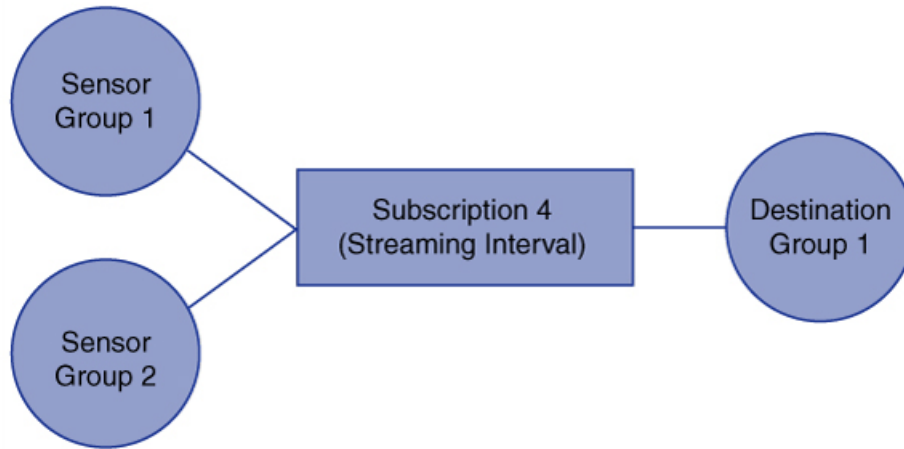


図 4: 単一の宛先グループにマッピングされている複数のセンサーグループ



355797

SAN Telemetry Streaming を構成するには、次の手順を実行します。

始める前に

- スイッチが Cisco MDS NX-OS リリース 8.3(1) 以降のリリースを実行していることを確認します。
- SAN アナリティクス機能を有効にします。「[SAN アナリティクスの有効化](#)」を参照してください。
- テレメトリ ソーススイッチのタイムゾーンが **clock** 構成コマンドで正しく設定されていることを確認します。そうしないと、SAN テレメトリ レシーバーは、受信した分析タイムスタンプを関連付けることができません。このコマンドの詳細については、『[Cisco MDS 9000 シリーズ コマンド リファレンス](#)』を参照してください。

手順

ステップ 1 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。

```
switch# configure terminal
```

ステップ 2 SAN Telemetry Streaming 機能を有効にします。

```
switch(config)# feature telemetry
```

ステップ 3 SAN Telemetry Streaming の構成モードに入ります。

```
switch(config)# telemetry
```

ステップ 4 (オプション) 既存の SSL または TLS 証明書を使用します。

```
switch(config-telemetry)# certificate certificate_path host_name
```

(注) Cisco MDS 9700 シリーズ スイッチでは、セキュアなテレメトリ構成のために、アクティブ スーパーバイザとスタンバイ スーパーバイザの両方でクライアント証明書を使用できることを確認してください。そうしないと、アップグレードまたはダウングレード後に SAN Telemetry Streaming が失敗します。 **copy bootflash:<client certificate file> bootflash://sup-standby/<client certificate file>** コマンドを使用して、アクティブ スーパーバイザからスタンバイ スーパーバイザにクライアント証明書をコピーします。

ステップ 5 (オプション) 宛先プロファイル構成モードを開始し、gRPC 転送プロトコルの送信の再試行の詳細を指定します。

1. switch(config-telemetry)# **destination-profile**
2. switch(conf-tm-dest-profile)# **use-retry size buffer_size**

宛先プロファイルには、たとえば、すべての宛先に固有のトランスポート リトライ バッファ サイズなどのパラメータを構成できます。

(注) バッファ サイズは MB 単位で、範囲は 10 ~ 1500 です。

ステップ 6 ID を持つセンサー グループを作成し、センサー グループ コンフィギュレーション モードを開始します。

```
switch(conf-tm-dest-profile)# sensor-group id
```

センサー グループは 1 つ以上のセンサー パスの集合です。

現在は、数字のセンサー グループ ID 値のみサポートされています。センサー グループでは、テレメトリ レポートの監視対象ノードを定義します。

ステップ 7 センサー グループにセンサー パスを追加します。

```
switch(conf-tm-sensor)# path sensor_path
```

sensor_path では、ストリーミングされる特定のインターフェイスの統計情報とプッシュ クエリが指定されます。1 つのセンサー グループに複数のセンサー パスを設定できます。テレメトリのセンサー パスは **path analytics: query_name**、インターフェイスの統計情報ストリーミングのセンサー パスは **path show_stats_fc slot/port** です。

(注) センサー パスの構文は設定時には検証されません。センサー パスが正しくないと、データ ストリーミングが失敗する可能性があります。

ステップ 8 宛先グループを作成して、宛先グループ コンフィギュレーション モードを開始します。

```
switch(conf-tm-sensor)# destination-group id
```

現在は、宛先グループ ID は数字の ID 値のみサポートしています。

(注) 宛先グループは 1 つ以上の宛先の集合です。

ステップ 9 発信データの宛先プロファイルを作成します。

```
switch(conf-tm-dest)# {ip | ipv6} address address port number [ protocol procedural-protocol encoding encoding-protocol]
```

(注) Cisco MDS NX-OS リリース 8.3(2) の時点では、gRPC がサポートされている唯一の転送プロトコルです。また、GPB とコンパクト GPB エンコーディングだけがサポートされています。

宛先グループがサブスクリプション ノードにリンクされている場合、テレメトリ データは、宛先プロファイルで指定されている IP アドレスとポートに送信されます。

ステップ 10 ID を持つサブスクリプション ノードを作成し、サブスクリプション 構成モードを開始します。

```
switch(conf-tm-dest)# subscription id
```

サブスクリプションは、センサー グループを宛先グループにマッピングします。

現在、サブスクリプション ID は数字の ID 値のみサポートしています。

ステップ 11 ID を持つセンサー グループをこのサブスクリプション ノードにリンクして、データ ストリーミングのサンプル間隔（ミリ秒単位）を設定します。

```
switch(conf-tm-sub)# snsr-grp id sample-interval interval
```

(注) 推奨の最小サンプル間隔は 30000 です。

現在、センサー グループ ID は数字の ID 値のみサポートしています。ストリーミング サンプル間隔の値を指定します。値はミリ秒単位である必要があります。サポートされているストリーミング サンプル間隔の最小値は 30000 ミリ秒です。最小値より大きい間隔値の場合、テレメトリ データが指定された間隔で定期的に送信される頻度に基づいたサブスクリプションが作成されます。

ステップ 12 ID を持つ宛先グループをこのサブスクリプションにリンクします。

```
switch(conf-tm-sub)# dst-grp id
```

現在は、宛先グループ ID は数字の ID 値のみサポートしています。

例：SAN テレメトリ ストリーミングの設定

この例では、ファイバチャネル インターフェイス 3/1 と 4/1 のデータを 30 秒ごとに、宛先 IP 1.2.3.4 ポート 50003 および IP 1:1::1:1 ポート 50009 にストリーミングし、test.pem を使用して検証された GPB エンコーディングを使用してストリームを暗号化するサブスクリプションの作成方法を示します。

```
switch# configure terminal
switch(config)# telemetry
switch(config-telemetry)# certificate /bootflash/test.pem foo.test.google.fr

switch(conf-tm-telemetry)# destination-group 100
switch(conf-tm-dest)# ip address 1.2.3.4 port 50003 protocol gRPC encoding GPB
```

例 : SAN テレメトリ ストリーミングの設定

```

switch(conf-tm-dest) # destination-group 1
switch(conf-tm-dest) # ipv6 address 1:1::1:1 port 50009 protocol gRPC encoding GPB-compact

switch(config-dest) # sensor-group 100
switch(conf-tm-sensor) # path show_stats_fc3/1
switch(conf-tm-sensor) # subscription 100
switch(conf-tm-sub) # snsr-grp 100 sample-interval 30000
switch(conf-tm-sub) # dst-grp 100

switch(config-dest) # sensor-group 1
switch(conf-tm-sensor) # path show_stats_fc4/1
switch(conf-tm-sensor) # subscription 1
switch(conf-tm-sub) # snsr-grp 1 sample-interval 30000
switch(conf-tm-sub) # dst-grp 1

```

この例では、30 秒ごとに **show** コマンドデータを収集し、レシーバ 1.2.3.4 と 1.1::1.1 に送信する定期的な収集の作成方法を示します。

```

switch# configure terminal
switch(config) # telemetry

switch(config-telemetry) # destination-group 100
switch(conf-tm-dest) # ip address 1.2.3.4 port 60001 protocol gRPC encoding GPB

switch(conf-tm-sensor) # destination-group 1
switch(conf-tm-dest) # ipv6 address 1:1::1:1 port 60009 protocol gRPC encoding GPB-compact

switch(config-dest) # sensor-group 100
switch(conf-tm-sensor) # subscription 100
switch(conf-tm-sub) # snsr-grp 100 sample-interval 30000
switch(conf-tm-sub) # dst-grp 100

switch(conf-tm-dest) # sensor-group 1
switch(conf-tm-sensor) # subscription 1
switch(conf-tm-dest) # snsr-grp 1 sample-interval 30000
switch(conf-tm-sub) # dst-grp 1

```

次に、センサーグループに複数のパスを含め、宛先グループに複数の宛先プロファイルを含め、サブスクリプションを複数のセンサーグループと宛先グループにリンクできる例を示します。

```

switch# configure terminal
switch(config) # telemetry

switch(config-telemetry) # sensor-group 100
switch(conf-tm-sensor) # path analytics:init
switch(conf-tm-sensor) # path analytics:inititl

switch(config-telemetry) # sensor-group 200
switch(conf-tm-sensor) # path analytics:inititl

switch(conf-tm-sensor) # destination-group 100
switch(conf-tm-dest) # ip address 1.2.3.4 port 50004
switch(conf-tm-dest) # ipv6 address 5:6::7:8 port 50005

switch(conf-tm-dest) # destination-group 200
switch(conf-tm-dest) # ip address 5.6.7.8 port 50001

```

```

switch(conf-tm-dest)# subscription 600
switch(conf-tm-sub)# snsr-grp 100 sample-interval 30000
switch(conf-tm-sub)# snsr-grp 200 sample-interval 30000
switch(conf-tm-sub)# dst-grp 100
switch(conf-tm-sub)# dst-grp 200

switch(conf-tm-dest)# subscription 900
switch(conf-tm-sub)# snsr-grp 200 sample-interval 30000
switch(conf-tm-sub)# dst-grp 100

```



- (注) *sensor_path* は、ストリーミングされる特定のインターフェイスの統計情報とプッシュクエリが指定される場所です。1つのセンサーグループに複数のセンサーパスを設定できます。テレメトリ ストリーミングのセンサーパスは **path analytics: query_name** であり、インターフェイスの統計情報ストリーミングの場合、センサーパスは **path show_stats_fc slot/port** です。センサーパスで指定されているクエリ名「*init*」、「*initit*」、および「*inititl*」は、SAN Analytics 機能で構成されています。詳細については、[プッシュクエリの設定](#)を参照してください。

次の例に、トランシーバストリーミングのサンプル構成を示します。

```

switch# configure terminal
switch(config)# telemetry

switch(config-telemetry)# sensor-group 200
switch(conf-tm-sensor)# path transceiver:fc1/1
switch(conf-tm-sensor)# path transceiver:fc13/1-48

switch(conf-tm-sensor)# show telemetry data collector details
-----
Row ID           Successful      Failed          Skipped          Sensor Path(GroupId)
-----
1                 398             14              0                show_stats_fc3/1-48(100)
2                30488           0                1                analytics:dcnmtgtITL(2)
3                 395             0                0                show_stats_fc5/1-48(100)
4                  0                0                0                transceiver:fc1/1(200)
5                  0                0                0                transceiver:fc13/1-48(200)
6                  0                0                0                analytics:dcnmtgtITN(1)

```

この例は、サンプル構成と、SAN Telemetry Streaming 構成を確認する方法を示しています。SAN Telemetry Streaming 構成を確認するために、**show telemetry data collector details** および **show telemetry transport session_id stats** コマンドの出力を確認することもできます。詳細については、[SAN テレメトリ ストリーミングの設定と統計情報の表示 \(14 ページ\)](#)を参照してください。

```

switch# configure terminal
switch(config)# telemetry

switch(config-telemetry)# destination-group 100
switch(conf-tm-dest)# ip address 1.2.3.4 port 50003 protocol gRPC encoding GPB
switch(conf-tm-dest)# ip address 1.2.3.4 port 50004 protocol gRPC encoding GPB

switch(config-telemetry)# destination-group 1
switch(conf-tm-dest)# ipv6 address 1:1::1:1 port 50008 protocol gRPC encoding GPB-compact

```

```

switch(conf-tm-dest)# ipv6 address 1:2::3:4 port 50009 protocol gRPC encoding GPB-compact

switch(conf-tm-dest)# end

switch# show running-config telemetry
!Command: show running-config telemetry
!Running configuration last done at: Thu Jun 14 08:14:24 2018
!Time: Thu Jun 14 08:14:40 2018
version 8.3(1)
feature telemetry
telemetry
destination-group 1
  ipv6 address 1:2::3:4 port 50008 protocol gRPC encoding GPB-compact
  ipv6 address 1:1::1:1 port 50009 protocol gRPC encoding GPB-compact
destination-group 100
  ip address 1.2.3.4 port 50003 protocol gRPC encoding GPB
  ip address 1.2.3.4 port 50004 protocol gRPC encoding GPB

```



(注) NPU の負荷は、アクティブな ITL と非アクティブな ITL の数を含む、すべての ITL に基づいています。したがって、NPU の負荷を確認する前に、クエリをクリアまたはページすることをお勧めします。

SAN テレメトリ ストリーミングの設定と統計情報の表示

次の Cisco NX-OS CLI **show** コマンドを使用して、SAN Telemetry Streaming 構成、統計情報、エラー、およびセッション情報を表示します。

この例は、SAN Telemetry Streaming の構成を反映している内部データベースを示します。

```

switch# show telemetry control database
Subscription Database size = 1
-----
Subscription ID      Data Collector Type
-----
100                  SDB

Sensor Group Database size = 1
-----
Row ID  Sensor Group ID  Sensor Group type  Sampling interval(ms)  Linked subscriptions
SubID
-----
1       100              Timer              /SDB                    30000                  /Running                1
100
Collection Time in ms (Cur/Min/Max): 53/9/81
Encoding Time in ms (Cur/Min/Max): 21/6/33
Transport Time in ms (Cur/Min/Max): 10470/1349/11036
Streaming Time in ms (Cur/Min/Max): 10546/9/11112

Collection Statistics:
  collection_id_dropped      = 0
  last_collection_id_dropped = 0
  drop_count                  = 0

```

Sensor Path Database size = 4

```
-----
Row ID  Subscribed Linked  Sec  Retrieve  Path  Query:  Filter
        Groups  Groups  level  (GroupId):
-----
1       No        1      0      Self     analytics:inititl(100): NA :  NA
GPB Encoded Data size in bytes (Cur/Min/Max): 162310/162014/162320
JSON Encoded Data size in bytes (Cur/Min/Max): 0/0/0

2       No        1      0      Self     show_stats_fc1/3(100): NA :  NA
GPB Encoded Data size in bytes (Cur/Min/Max): 2390/2390/2390
JSON Encoded Data size in bytes (Cur/Min/Max): 0/0/0

3       No        1      0      Self     analytics:initit(100): NA :  NA
GPB Encoded Data size in bytes (Cur/Min/Max): 158070/157444/158082
JSON Encoded Data size in bytes (Cur/Min/Max): 0/0/0

4       No        1      0      Self     analytics:init(100): NA :  NA
GPB Encoded Data size in bytes (Cur/Min/Max): 159200/158905/159212
JSON Encoded Data size in bytes (Cur/Min/Max): 0/0/0

```

Destination Group Database size = 1

> use-vrf : default

```
-----
Destination Group ID  Refcount
-----
100                    1

```

Destination Database size = 3

```
-----
Dst IP Addr      Dst Port  Encoding  Transport  Count
-----
10.30.217.80    50009    GPB       gRPC       1
2001:420:301:2005:3::11
                  60003    GPB       gRPC       1
2001:420:54ff:a4::230:e5
                  50013    GPB       gRPC       1

```

switch(conf-tm-dest)# **show telemetry control database sensor-groups**

Sensor Group Database size = 1

```
-----
Row ID Sensor Group ID Sensor Group type  Sampling interval(ms)  Linked subscriptions
SubID
-----

```

```
1       100                Timer /SDB           30000      /Running      1

```

100

Collection Time in ms (Cur/Min/Max): 53/9/81

Encoding Time in ms (Cur/Min/Max): 21/21/33

Transport Time in ms (Cur/Min/Max): 10304/461/15643

Streaming Time in ms (Cur/Min/Max): 10380/9/15720

Collection Statistics:

collection_id_dropped = 0

last_collection_id_dropped = 0

drop_count = 0



- (注) コマンド出力内の SDB は、SAN データ コレクターのタイプです。テレメトリは、他のサポート対象プラットフォームの DME、NX-API、および YANG データ ソースもサポートしていません。

この例は、SAN Telemetry Streaming の構成の内部データベースの統計を示します。

```
switch# show telemetry control stats
show telemetry control stats entered
```

Error Description	Error Count
Chunk allocation failures	0
Sensor path Database chunk creation failures	0
Sensor Group Database chunk creation failures	0
Destination Database chunk creation failures	0
Destination Group Database chunk creation failures	0
Subscription Database chunk creation failures	0
Sensor path Database creation failures	0
Sensor Group Database creation failures	0
Destination Database creation failures	0
Destination Group Database creation failures	0
Subscription Database creation failures	0
Sensor path Database insert failures	0
Sensor Group Database insert failures	0
Destination Database insert failures	0
Destination Group Database insert failures	0
Subscription insert to Subscription Database failures	0
Sensor path Database delete failures	0
Sensor Group Database delete failures	0
Destination Database delete failures	0
Destination Group Database delete failures	0
Delete Subscription from Subscription Database failures	0
Sensor path delete in use	0
Sensor Group delete in use	0
Destination delete in use	0
Destination Group delete in use	0
Delete destination(in use) failure count	0
Sensor path Sensor Group list creation failures	0
Sensor path prop list creation failures	0
Sensor path sec Sensor path list creation failures	0
Sensor path sec Sensor Group list creation failures	0
Sensor Group Sensor path list creation failures	0
Sensor Group Sensor subs list creation failures	0
Destination Group subs list creation failures	0
Destination Group Destinations list creation failures	0
Destination Destination Groups list creation failures	0
Subscription Sensor Group list creation failures	0
Subscription Destination Groups list creation failures	0
Sensor Group Sensor path list delete failures	0
Sensor Group Subscriptions list delete failures	0
Sensor Group Subscriptions unsupported data-source failures	0
Destination Group Subscriptions list delete failures	0
Destination Group Destinations list delete failures	0
Subscription Sensor Groups list delete failures	0
Subscription Destination Groups list delete failures	0
Destination Destination Groups list delete failures	0
Failed to delete Destination from Destination Group	0
Failed to delete Destination Group from Subscription	0

```
Failed to delete Sensor Group from Subscription      0
Failed to delete Sensor path from Sensor Group     0
Failed to get encode callback                      0
Failed to get transport callback                   0
```

次に、データ収集に関する簡単な統計情報の例を示します。

```
switch# show telemetry data collector brief
```

Row ID	Collector Type	Successful	Failed	Skipped
1	NX-API	0	0	0
2	SDB	1513	902	0

この例では、すべてのセンサーパスの詳細を含む、データ収集に関する詳細な統計情報を示します。

```
switch# show telemetry data collector details
```

Row ID	Successful	Failed	Skipped	Sensor Path(GroupId)
1	496	305	0	analytics:inititl(100)
2	16	0	0	show_stats_fcl/3(100)
3	507	294	0	analytics:initit(100)
4	498	303	0	analytics:init(100)



(注) 出力でスキップされた数は、ゼロ差分レコードがフェッチされた回数を示します。

この例では、SAN Telemetry Streaming パイプラインの統計情報を表示します。SAN Telemetry Streaming パイプラインからは、収集キューと転送キューの統計情報（キューサイズ、キューのドロップ数など）が得られます。

```
switch# show telemetry pipeline stats
Main Statistics:
  Timers:
    Errors:
      Start Fail      =    0

  Data Collector:
    Errors:
      Node Create Fail =    0

  Event Collector:
    Errors:
      Node Create Fail =    0   Node Add Fail   =    0
      Invalid Data     =    0

  Memory:
    Allowed Memory Limit = 838860800 bytes
    Occupied Memory      = 53399552 bytes

Queue Statistics:
```

```

Request Queue:
  High Priority Queue:
    Info:
      Actual Size      = 50    Current Size      = 0
      Max Size         = 0     Full Count        = 0

    Errors:
      Enqueue Error    = 0     Dequeue Error     = 0

  Low Priority Queue:
    Info:
      Actual Size      = 50    Current Size      = 0
      Max Size         = 0     Full Count        = 0

    Errors:
      Enqueue Error    = 0     Dequeue Error     = 0

Data Queue:
  High Priority Queue:
    Info:
      Actual Size      = 160000  Current Size      = 0
      Max Size         = 0     Full Count        = 0

    Errors:
      Enqueue Error    = 0     Dequeue Error     = 0

  Low Priority Queue:
    Info:
      Actual Size      = 2     Current Size      = 0
      Max Size         = 0     Full Count        = 0

    Errors:
      Enqueue Error    = 0     Dequeue Error     = 0

```

この例は、構成されているすべての転送セッションを示します。

```

switch# show telemetry transport

Session Id      IP Address      Port      Encoding  Transport  Status
-----
2               10.30.217.80   50009    GPB       gRPC       Connected
0               2001:420:301:2005:3::11
                60003         GPB       gRPC       Connected
1               2001:420:54ff:a4::230:e5
                50013         GPB       gRPC       Transmit Error
-----

Retry buffer Size:          10485760
Event Retry Messages (Bytes): 0
Timer Retry Messages (Bytes): 10272300
Total Retries sent:        0
Total Retries Dropped:     5377

```

次に、特定の転送セッションの詳細なセッション情報の例を示します。

```

switch# show telemetry transport 0

Session Id:          2
IP Address:Port      10.30.217.80:50009
Transport:           GRPC
Status:              Connected

```

```

Last Connected:      Fri Jun 22 07:07:12.735 UTC
Last Disconnected:  Never
Tx Error Count:      0
Last Tx Error:       None
Event Retry Queue Bytes:  0
Event Retry Queue Size:  0
Timer Retry Queue Bytes:  0
Timer Retry Queue Size:  0
Sent Retry Messages:  0
Dropped Retry Messages:  0

```

次に、特定の転送セッションの詳細の例を示します。

```

switch# show telemetry transport 2 stats

Session Id:          2
Connection Stats
  Connection Count    2
  Last Connected:     Fri Jun 22 07:07:12.735 UTC
  Disconnect Count    0
  Last Disconnected:  Never
Transmission Stats
  Compression:        disabled
  Source Interface:   not set()
  Transmit Count:     44
  Last TX time:       Fri Jun 22 07:14:16.533 UTC
  Min Tx Time:        227 ms
  Max Tx Time:        3511 ms
  Avg Tx Time:        1664 ms
  Cur Tx Time:        227 ms

```

次のコマンドでは、特定の転送セッションの詳細なエラーの統計情報が表示されます。

```

switch# show telemetry transport 2 errors

Session Id:          1
Connection Errors
  Connection Error Count:  0
Transmission Errors
  Tx Error Count:         1746
  Last Tx Error:          Fri Jun 22 07:15:07.970 UTC
  Last Tx Return Code:    UNAVAILABLE

```



(注) 以下は、**show telemetry transport errors** コマンド出力の戻りコードの説明です。

- OK : エラーは検出されませんでした。
- UNAVAILABLE : 設定された IP アドレスまたはポートに到達できません。設定をチェックして、正しい IP アドレスまたはポートが設定されていることを確認します。
- DEADLINE_EXCEEDED : レシーバからの応答が 30 秒以上ないか、ネットワークの遅延が発生しています。

SAN テレメトリ ストリーミングのトラブルシューティング

show tech-support telemetry コマンドを使用して、トラブルシューティング用のテレメトリ データを収集します。エラーを発見した場合は、[SAN テレメトリ ストリーミングの設定 \(8 ページ\)](#) をチェックして構成を確認します。

テレメトリ ステータスのトラブルシューティングには、次の情報を使用します。

1. **show analytics system-load** コマンドを使用して、NPU の負荷を確認します。NPU の負荷が高い場合は、一部のポートでの分析を無効にします。

```
switch# show analytics system-load
n/a - not applicable
----- Analytics System Load Info -----
| Module | NPU Load (in %) | ITLs | ITNs | Both | Hosts | Targets |
|         | SCSI NVMe Total | SCSI | NVMe | Total | SCSI | NVMe | Total |
-----+-----+-----+-----+-----+-----+-----+-----+
| 1       | 0 0 0          | 0    | 0    | 0    | 0    | 0    | 0    | |
| 4       | 64 0 64       | 20743| 0    | 20743| 0    | 0    | 346  |
| 5       | 0 0 0         | 0    | 0    | 0    | 0    | 0    | 0    |
| 8       | 0 0 0         | 0    | 0    | 0    | 0    | 0    | 0    |
| 12      | 0 12 12       | 0    | 300  | 300  | 0    | 0    | 40   |
| 13      | 0 0 0         | 0    | 0    | 0    | 0    | 0    | 0    |
| 18      | 0 13 13       | 1    | 1    | 2    | 1    | 2    | 0    |
| Total   | n/a n/a n/a   | 20744| 301  | 21045| 1    | 1    | 2    | 346  |
|         |         |         |         |         |         |         |         |
-----+-----+-----+-----+-----+-----+-----+
As of Mon Apr 1 05:31:10 2019
```

2. **show telemetry control database sensor-groups** コマンドを使用し、コマンド出力でサンプルインターバルタイマーが動作しているかどうかを確認します。タイマーが動作していない場合は、タイマーが正しく設定されているかどうかを確認します。

```
switch# show telemetry control database sensor-groups
Sensor Group Database size = 3
-----
Row ID      Sensor Group ID  Sensor Group type  Sampling interval(ms)  Linked
subscriptions  SubID
-----+-----+-----+-----+-----+-----+
1          100              Timer /SDB           5000                   /Running                1
          100
Collection Time in ms (Cur/Min/Max): 0/0/1
Encoding Time in ms (Cur/Min/Max): 0/0/0
Transport Time in ms (Cur/Min/Max): 0/0/0
Streaming Time in ms (Cur/Min/Max): 1/1/4753

Collection Statistics:
  collection_id_dropped      = 0
  last_collection_id_dropped = 0
  drop_count                 = 0

2          1          Timer /SDB           30000                  /Running                1
          1
Collection Time in ms (Cur/Min/Max): 5/4/16
Encoding Time in ms (Cur/Min/Max): 2/2/11
Transport Time in ms (Cur/Min/Max): 644/635/1589
Streaming Time in ms (Cur/Min/Max): 3223/3168/4964

Collection Statistics:
  collection_id_dropped      = 0
```

```
last_collection_id_dropped = 0
drop_count                  = 0
```

3. **show telemetry data collector details** コマンド使用し、コマンド出力でデータの収集にエラーがあるかどうかを確認します。エラーを発見した場合は、SAN Telemetry Streaming の構成時に指定した *sensor_path* が誤っているため修正する必要があります。

```
switch# show telemetry data collector details
```

```
-----
Row ID      Successful  Failed Skipped  Sensor Path(GroupId)
-----
1           0           2994    0        analytics:panup(1)
2          2994           0     0        show_stats_fc2/2(1)
3           0           2994    0        analytics:port(1)
4          2994           0     0        show_stats_fc2/6(1)
5          2994           0     0        show_stats_fc2/1(1)
```

4. **show logging logfile | grep -i telemetry** コマンドを使用して、Syslog メッセージ内のエラーを確認します。

```
switch# show logging logfile | grep -i telemetry
2018 Jun 28 16:26:17 switch %TELEMETRY-4-TRANSPORT_SEND_ERROR: GRPC send to
172.20.30.129:60002 failed. (DEADLINE_EXCEEDED(len:2876013))
```

5. **ステップ 1**、**ステップ 2**、および**ステップ 3** で使用して問題が見つからなかった場合、問題は転送プロトコルにあることが考えられます。**show telemetry transport 0 errors** コマンドを使用して、コマンド出力で転送プロトコルにエラーがあるかどうかを確認します。

次の理由が原因で転送プロトコルエラーが発生することがあります。

- 宛先プロファイルまたはサブスクリプションで、間違った IP アドレスまたはポートを設定している。宛先プロファイルまたはサブスクリプションの IP アドレスまたはポートを修正します。
- レシーバが起動していない。レシーバがアクティブになっていて、gRPC ポートをリスンしていることを確認します。
- レシーバは起動しているが、メッセージを処理していない。レシーバアプリケーションのエラーを確認します。
- 管理 IP の問題がある。telnet コマンドを使用して、IP アドレスとポートに到達可能かどうかをテストします。

```
switch# show telemetry transport 1 errors
```

```
Session Id:                1
Connection Errors
  Connection Error Count:   0
Transmission Errors
  Tx Error Count:          0
  Last Tx Error:           None
```

Last Tx Return Code: OK

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。