

カスタムPromQLクエリーによるSMFコールフロー問題のトラブルシューティング

内容

[概要](#)

[省略形](#)

[SMFコールフローの問題をトラブルシューティングするためにクエリをカスタマイズする理由](#)
[GrafanaとPrometheus](#)

[Grafana](#)

[Prometheus](#)

[PromQLクエリ](#)

[ダッシュボードとパネルの作成方法](#)

[例：カスタマイズされたクエリーとグラフを使用したトラブルシューティング](#)

概要

このドキュメントでは、コールフローに関連する問題をトラブルシューティングするために、Cisco SMFでGrafana/Prometheusを使用してカスタムクエリーを作成する方法について説明します。

省略形

SMF セッション管理機能
UDM ユニファイドデータ管理
AMF アクセスおよびモビリティ機能
PDU プロトコル データ ユニット

SMFコールフローの問題をトラブルシューティングするためにクエリをカスタマイズする理由

組み込みのダッシュボードは、重要なKPIとノードのヘルス統計に関する優れたグラフを提供しますが、通常の問題シナリオのトラブルシューティングにPromQLクエリとGrafanaの可能性を最大限に活用するために、カスタムクエリが重要です。独自のpromqlクエリとグラフを使用すると、特定の障害を切り分けるための汎用性と利便性が向上します。

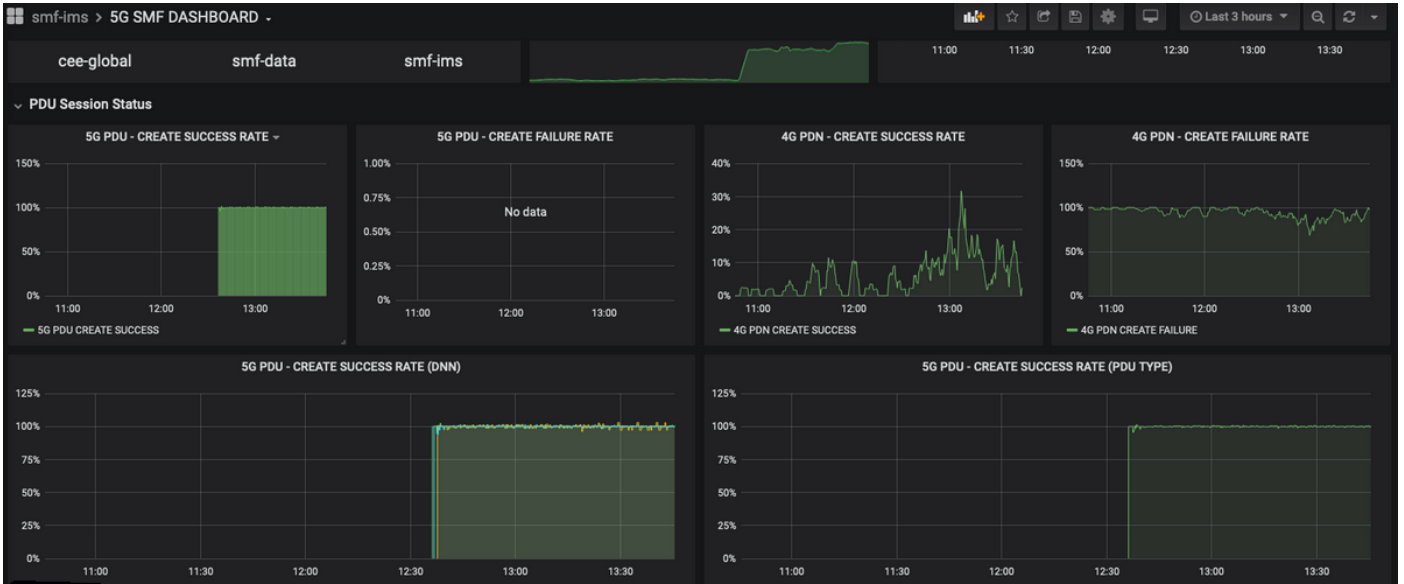
内蔵ダッシュボードの利点：

- Grafanaは、SMF統計情報を参照するための使いやすいグラフィカルインターフェイスを提供します。
- ほとんどのKPIと統計情報を確認できる組み込みのグラフダッシュボードがあります。

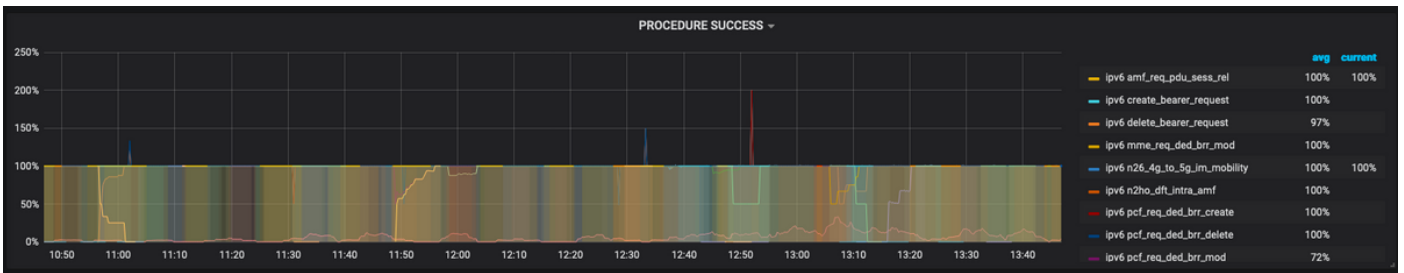
例：

5G SMFダッシュボード

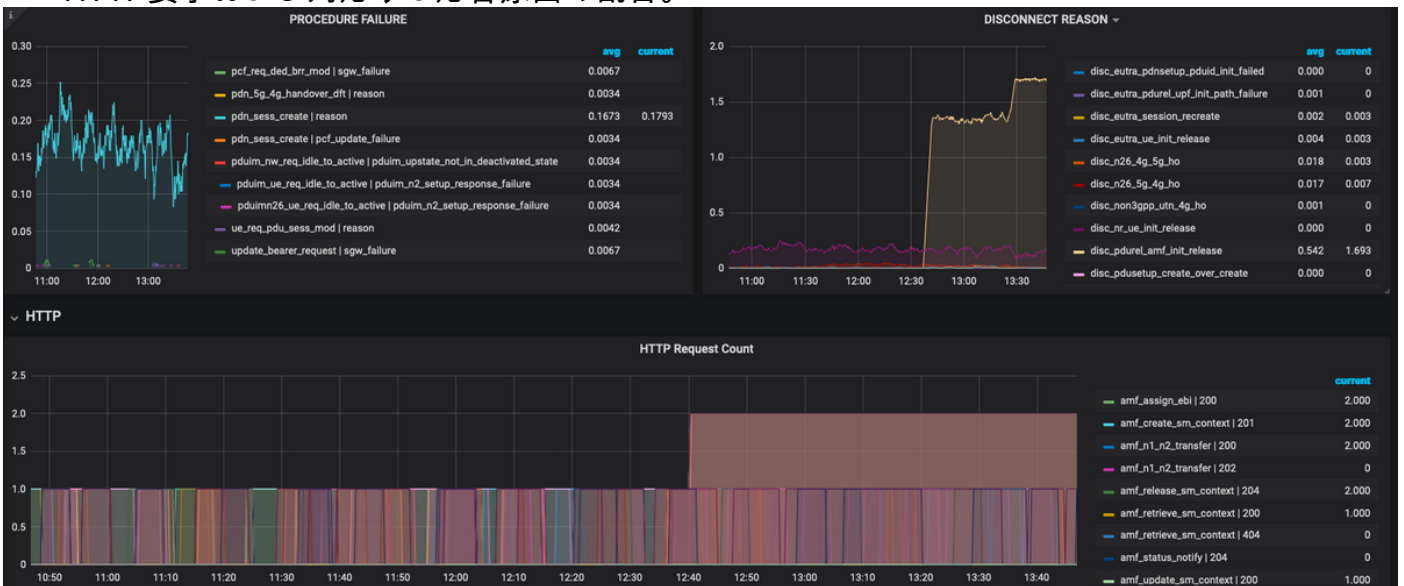
- 5G PDU Create Failure/Success Rate
- 4G PDN Create Failure/Success Rate



- プロシージャごとの成功率



- プロシージャごとのエラーの原因の割合。
- 切断理由のパーセンテージ。
- HTTP要求および対応する応答原因の割合。



さらにトラブルシューティングするには、次の手順を実行します。

- 使用可能なダッシュボードとパネルは、主にパーセンテージとKPIに関するものです。さらに

詳しく調査する際には、この障害を引き起こした特定のシナリオとメッセージを特定するために、詳細を調べる必要があります。

- 特定の正規表現を使用してカスタマイズされたクエリは、これらの統計情報を関連付け、トリガーを切り分けるのに役立ちます。
- これらのクエリは、SMFのgrafanaまたはオフラインのgrafanaでグラフをプロットするために使用できます。また、tac-debugパッケージからのメトリックダンプも使用できます。
- さまざまなサービスに関連付けられたメトリックの範囲を使用でき、特定のシナリオをトラブルシューティングするためにラベルキー/値ペアを使用してフィルタリングすることもできます。

グラフィナとプロメテウス

グラフィナ

「Grafanaはオープンソースの可視化と分析ソフトウェアです。メトリックのクエリー、視覚化、アラートの実行、および保存場所に関係なくメトリックの調査が可能です」

Cisco SMFは、内蔵のGrafanaを使用して、アプリケーションコンテナからのリアルタイム統計データをプロットします。

プロメテウス

Prometheusは、メトリック名とキー/値ペアで識別される時系列データと、これらのデータにアクセスするためのPromQLという柔軟なクエリ言語を備えた多次元データモデルを提供します。

Prometheusは、マイクロサービスから統計情報/カウンタを収集するために使用されます。

メトリック：時系列の統計情報の識別子です。

ラベル：メトリックはラベルで構成されます。キーと値のペアは基本的にどれか？特定のメトリックのラベルの組み合わせは、時系列データの特定のインスタンスを識別します

例：

```
smf_service_stats{app_name="SMF",cluster="Local",data_center="DC",dnn="intershat",emergency_call="false",instance_id="0",pdu_type="ipv4",procedure_type="pdu_sess_create",qos_5qi="",rat_type="",reason="",service_name="smf-service",status="attempted",up_state=""} 43
```

緑色で強調表示されているメトリック「smf_service_stats」には、黄色で強調表示されている多数のラベルがあります。

これらのラベルキー/値ペアを使用して、特定のデータ系列を選択できます。

PromQLクエリ

PrometheusはPromQLと呼ばれる機能クエリ言語を提供します。組み込み関数はPromQLで利用可能です(例: Sum(), by(), count()など)を使用すると、グラフ形式または表形式で特定の時系列データを選択できます。

例:

```
sum(smf_service_stats{status="success"}) by (procedure_type)
```

この例では、smf_service_statsメトリックからprocedure_typeでデータを選択します。ここで、status = "success"

合計 (デイメンションの合計を計算)

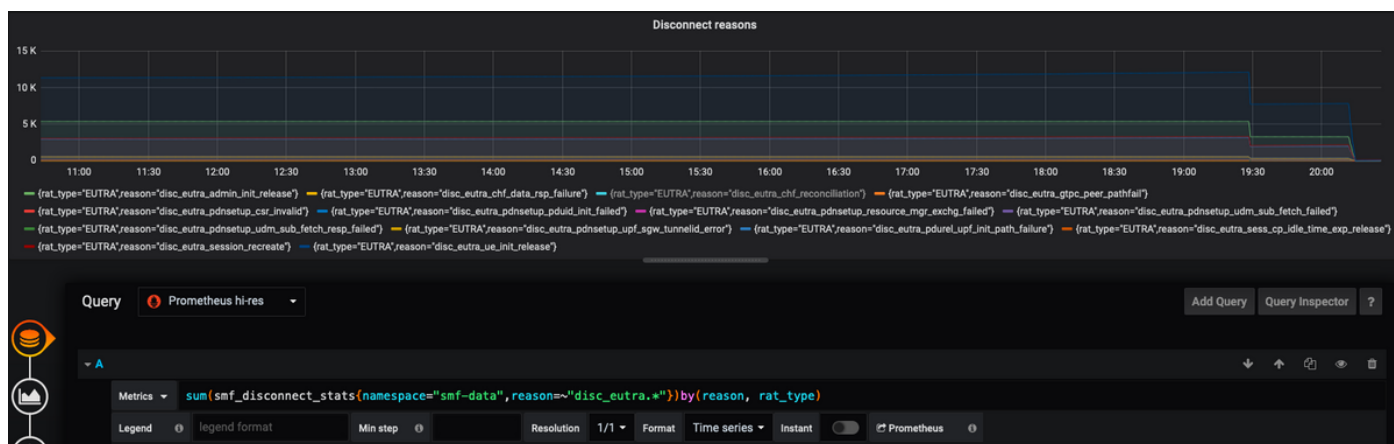
by (出力をラベルでグループ化)

フィルタは、ラベルキー/値ペアを使用して合計の内部で使用し、グラフをさらにフィルタできます。

例 1:

```
sum(smf_disconnect_stats{namespace="smf-data", reason=~"disc_eutra.*"}) by (reason, rat_type)
```

ここで名前空間smf-dataが選択され、その理由として、disc_eutraで始まるすべての切断理由 (つまり4Gの切断理由) を考慮します。



例 2:

```
sum(smf_restep_http_msg{namespace="smf-data", api_name=~"sdm.*"}) by (api_name, message_direction, response_status, response_cause)
```

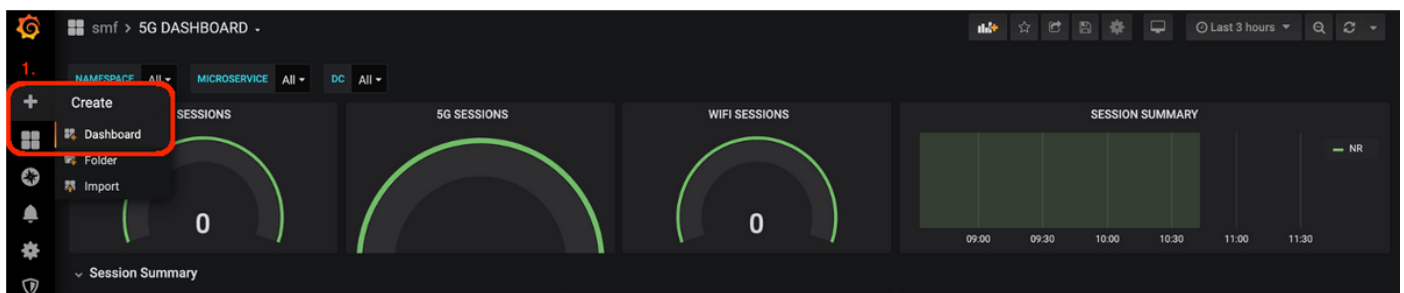
このクエリは、SMF - UDM sdm-subscriptionメッセージを応答原因とともに出力します。



ダッシュボードとパネルの作成方法

新しいダッシュボードを追加します。

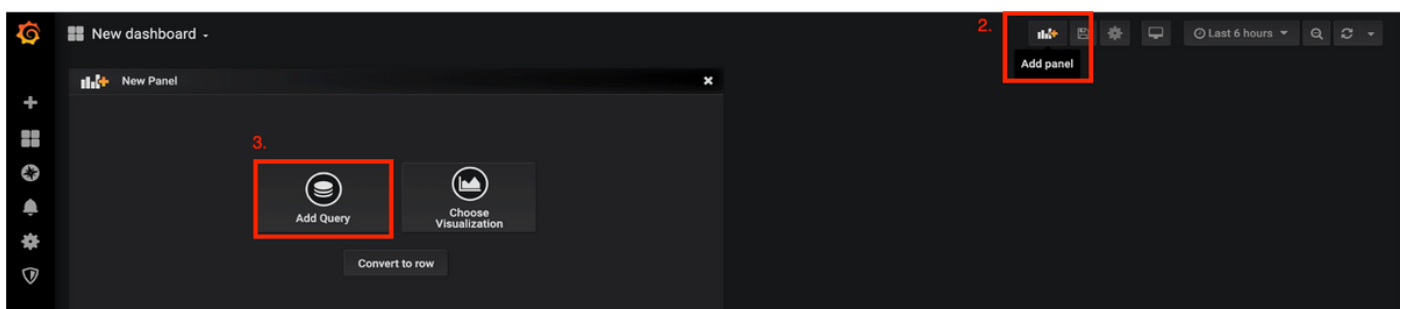
ステップ1：次の図に示すように、[Create] > [Dashboard]に移動します。



[New Panel- Add Query]を追加します。

ステップ2：新しいパネルを追加するには、上部の[Add Panel]オプションに移動します。

ステップ3:[Add Query]ボタンを選択します。

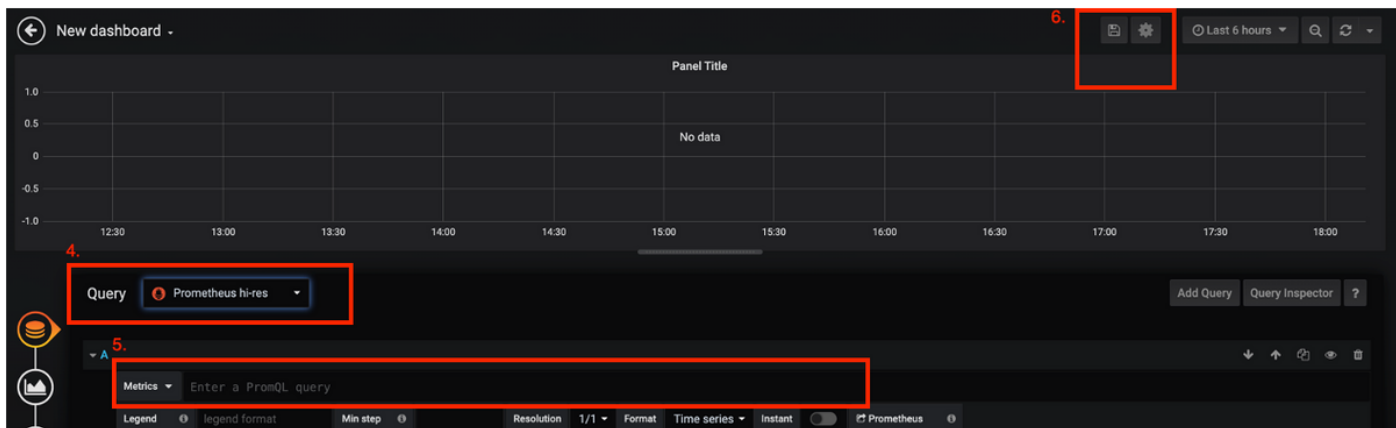


[Query Type- Prometheus hi-res]を選択します。

ステップ4:[クエリ]ドロップダウンリストの[Prometheus hi-res]オプションを選択します。

ステップ5：次に、指定されたボックスにpromqlクエリを追加します。

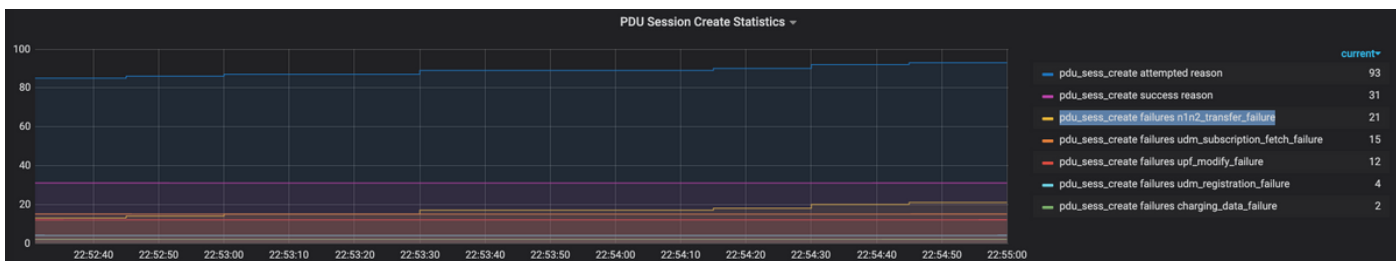
ステップ6：パネルを保存します。



例：カスタマイズされたクエリーとグラフを使用したトラブルシューティング

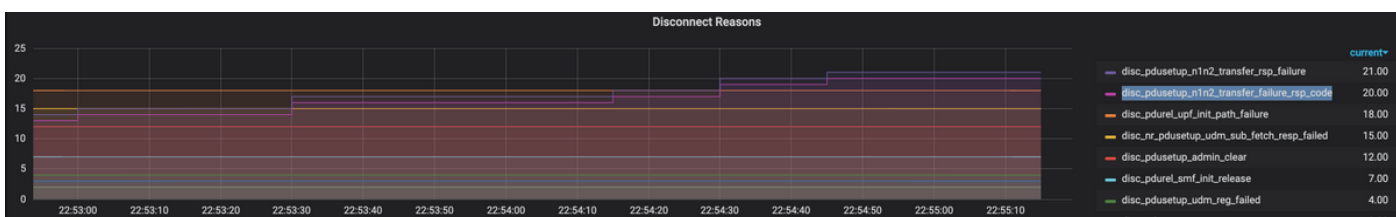
PDUセッション確立の失敗 – N1N2応答障害

ステップ1: KPIデット観測とPDUセッションの作成に失敗したことを特定します。



```
Query: sum by (procedure_type, pdu_type, status, reason)
(smfservice_stats{namespace="smf", procedure_type="pdu_sess_create"})
```

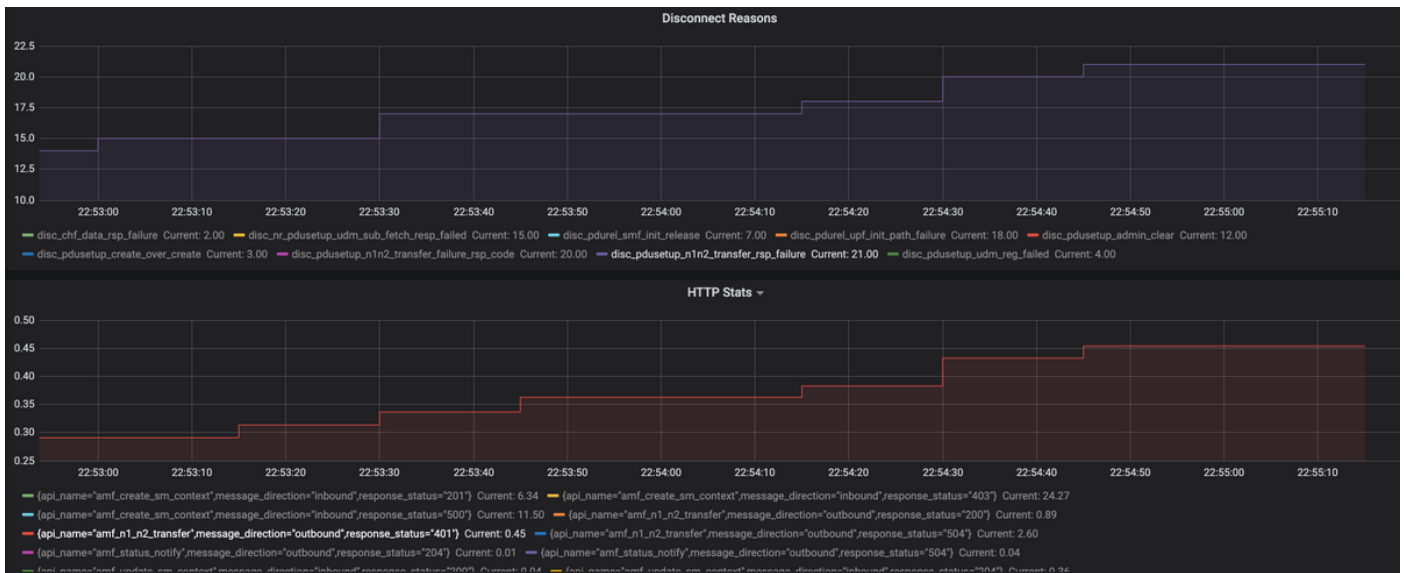
ステップ2: 障害の原因は「n1n2_transfer_failure_rsp_code」です。接続解除の原因を見てみましょう。



```
Query: sum(smfservice_stats{namespace="smf"}) by (reason)
```

ステップ3: 切断理由「disc_pdusetup_n1n2_transfer_rsp_failure」は、AMFピアからの否定応答を示します。SMF-AMFインタラクションはHTTPサービススペースのインターフェイス上にあるため、HTTP統計情報をさらに調べる必要があります(メトリック: smf_restep_http_msg)

HTTP統計情報は、障害中にSMFがHTTPステータスコード401 - Unauthorized from AMFを受信したことを示します



Query: sum(smf_restep_http_msg{namespace="smf"}) by(api_name,message_direction,response_status)
 トラブルシューティングに必要な重要なメトリック：

- smf_disconnect_stats
- smf_proto_pfcpc_msg_total
- smf_service_stats
- smf_restep_http_msg
- smf_n1_message_stats
- smf_proto_pfcpc_msg_total
- nodemgr_msg_stats
- nodemgr_gtpc_msg_stats
- chf_message_stats
- policy_msg_processing_status
- procedure_protocol_total
- procedure_service_total

[SMFメトリックの詳細：](#)

これらの例で示されているように、特定の障害シナリオで必要に応じて独自のカスタムグラフをプロットし、さまざまなメッセージを関連付けて障害を切り分けることができます。このようなクエリは、Tac_debug_pkgからのメトリックデータがローカルグラフアナにマウントされた後も、ローカルシステムで実行できます。