

在Nexus平台上配置并验证遥测

目录

[简介](#)

[先决条件](#)

[要求](#)

[使用的组件](#)

[拓扑](#)

[Nexus到接收方的连接](#)

[背景信息](#)

[传统轮询与流遥测](#)

[实际使用案例](#)

[遥测处理概述](#)

[遥测架构和术语](#)

[数据收集](#)

[数据管理引擎 — DME](#)

[受管对象和管理信息树](#)

[可分辨名称 — DN](#)

[传感器路径](#)

[数据编码](#)

[数据传输](#)

[遥测接收器](#)

[目标组](#)

[传感器组](#)

[订阅](#)

[定期和基于事件的遥测](#)

[定期遥测勘测](#)

[基于事件的遥测](#)

[实验中使用的收集行为](#)

[准备遥测接收器](#)

[本实验中使用的接收器](#)

[在NX-OS上配置流遥测](#)

[启用遥测功能](#)

[配置目标组](#)

[配置传感器组1](#)

[配置订阅1](#)

[配置传感器组2](#)

[配置订阅2](#)

[配置环回接口](#)

[配置传感器组3](#)

[配置订阅3](#)

[完成初始遥测配置](#)

[验证定期遥感勘测](#)

- [验证遥测传输](#)
- [检验周期性传感器组](#)
- [验证数据收集](#)
- [在接收方检验定期数据](#)
- [检验输出和运行信息](#)
- [定期遥测验证摘要](#)

[验证基于事件的遥测](#)

- [验证基于事件的订用](#)
- [生成基于事件的更新](#)
- [验证初始快照和事件消息](#)
- [验证事件收集统计信息](#)
- [验证事件收集器错误](#)
- [基于事件的遥测验证摘要](#)

[基本故障排除](#)

- [验证当前传输状态](#)
- [检查数据收集计数器](#)
- [确定跳过收集的原因](#)
- [检查传输错误](#)
- [验证传输恢复](#)
- [检查配置和事件收集器错误](#)
- [故障排除工作流程](#)

[实际应用示例：监控系统资源](#)

- [配置系统资源遥测](#)
- [查看资源路径标签](#)
- [验证系统资源遥测](#)
- [请比较NX-OS系统资源](#)
- [在接收方验证资源数据](#)
- [观察一段时间内的资源利用率](#)

[结论](#)

简介

本文档介绍如何在运行Cisco NX-OS的Cisco Nexus 9000交换机上实施和验证流遥测。

先决条件

要求

Cisco 建议您具有以下主题的基础知识：

- 思科NX-OS CLI
- 基本第3层IP连接
- 基本网络监控概念
- 基本熟悉网络遥测概念

使用的组件

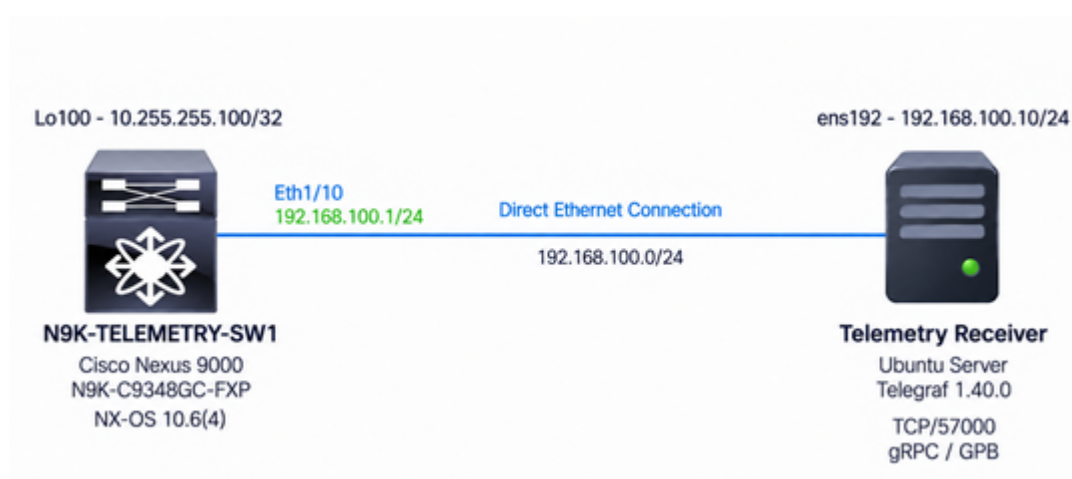
本文档中的信息基于以下软件和硬件版本：

组件	平台/软件	版本/值	目的
N9K-TELEMETRY-SW1	N9K-C9348GC-FXP	10.6(4)	遥测源
遥测接收器	Ubuntu服务器	22.04.5	外部遥测接收器
遥测应用	电报	1.40.0	Google Protocol Buffers(GPB)-over-gRPC遥测接收器
侦听端口	TCP	57000	遥测接收器端口
接收器输出格式	JavaScript对象表示法(JSON)	—	可人工读取的遥测输出

本文档中的信息都是基于特定实验室环境中的设备编写的。本文档中使用的所有设备最初均采用原始（默认）配置。如果您的网络处于活动状态，请确保您了解所有命令的潜在影响。

拓扑

本实验使用直接连接到Ubuntu服务器的Cisco Nexus 9000交换机，该服务器作为外部遥测接收器运行。



Nexus交换机和遥测接收器通过Ethernet1/10和ens192直接连接。

Nexus到接收方的连接

Ethernet1/10在Nexus交换机和Ubuntu接收器之间提供第3层连接。

接口配置为：

```
N9K-TELEMETRY-SW1# show running-config interface ethernet1/10 interface Ethernet1/10 description TELEMETRY-COLLECTOR ip address 192.168.100.1/24 no shutdown
```

The Ubuntu receiver uses: Interface: ens192 IP address: 192.168.100.10/24

背景信息

网络监控和可视性是了解网络运行状况、识别异常行为和排除网络问题的基础。

Cisco NX-OS提供多种机制来从Nexus交换机检索或导出操作信息，包括CLI、简单网络管理协议(SNMP)和系统日志。传统监控工作流程（如SNMP轮询和基于CLI的收集）通常依赖拉式模型，在这种模型中，外部监控系统定期从网络设备请求信息。

流遥测引入推送模型，其中网络设备根据配置的订用向外部接收器发送选定的运行数据。这提供了一种结构化机制来收集网络信息，而无需监控系统持续轮询设备。

传统轮询与流遥测

在传统的轮询模式中，监控系统确定何时从网络设备检索信息。

例如，网络管理系统(NMS)可以每60秒请求一次接口统计信息。这样可提供设备状态的定期快照；但是，监控应用的可见性与配置的轮询间隔直接相关。

通过流遥测，Nexus交换机将根据配置的订用向遥测接收器发送所选信息。

基本区别可以总结如下：

传统轮询	流遥测
客户端发起请求。	网络设备发送数据。
拉模型	推送模型
根据轮询间隔检索数据。	数据根据遥测订阅发送。
通常提供定期快照。	支持定期和基于事件的收集。
监控系统向设备请求信息。	设备向接收器传输所选信息。

流遥测不一定替代传统的监控机制。CLI、SNMP、系统日志和遥测可以共存，并服务于不同的操作目的。

主要区别在于数据收集模型。

实际使用案例

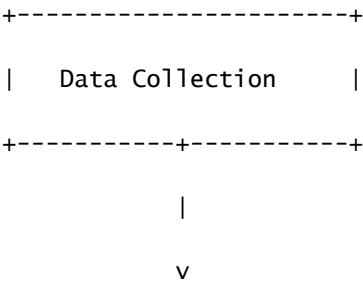
在生产环境中，流遥测通常用于提供对网络和系统运行状况的持续可视性。

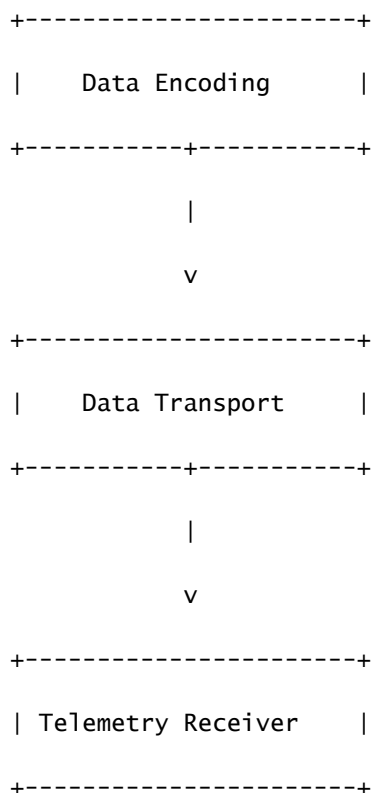
典型使用案例包括监控接口计数器和状态更改、系统资源（如CPU和内存利用率）以及数据中心交换矩阵信息(如虚拟可扩展局域网(VXLAN)对等体和边界网关协议(BGP)对等体状态)。基于事件的遥测还可用于报告发生更改时的运行状态更改。

导出的遥测数据随后可由外部监控和可观察平台用于控制面板、警报、历史分析和故障排除。

遥测处理概述

从较高层面来说，NX-OS上的流遥测可理解为四个主要阶段：





这些阶段回答了四个基本问题：

- 数据收集:收集哪些信息？
- 数据编码:如何表示信息？
- 数据传输：遥测数据如何传输？
- 遥测接收器：遥测数据发送到哪里？

遥测架构和术语

数据收集

第一阶段确定必须从交换机收集哪些信息。

对于本文档中的示例，遥测数据是从数据管理引擎(DME)收集的。

数据管理引擎 — DME

数据管理引擎在NX-OS中维护配置和操作信息的结构化表示。

DME不是将设备信息仅表示为CLI文本，而是将信息组织为可通过分层路径访问的托管对象。

例如：

```
sys/intf/phys-[eth1/10]
```

标识与Ethernet1/10关联的DME对象。

本实验中使用的其他DME路径包括：

```
sys/intf/phys-[eth1/10]/dbgIfIn
sys/intf/phys-[eth1/10]/dbgIfOut
sys/intf/phys-[eth1/10]/phys
sys/intf/lb-[lo100]
```

每条路径代表一个不同的对象或通过DME可用的信息部分。这些路径与整个实验配置中已使用的路径相同。

受管对象和管理信息树

DME数据库由托管对象(MO)组成。

受管对象表示NX-OS管理模型中的实体，例如：

- 物理接口
- 环回接口
- 接口输入统计信息
- 接口输出统计信息

托管对象在管理信息树(MIT)中按层次进行组织。

本实验中所用对象的简化表示如下：

```
sys
|
+-- intf
|
```

```
+-- phys-[eth1/10]
| |
| +-- dbgIfIn
| +-- dbgIfOut
| +-- phys
|
+-- lb-[lo100]
```

此层次结构有助于了解遥测传感器路径如何识别DME中的特定信息。

可分辨名称 — DN

每个受管对象都可以由可分辨名称(DN)唯一标识。

DN表示从DME树的根到目标对象的分层路径。

For example: sys/intf/lb-[lo100]

标识Loopback100托管对象。

类似地：

sys/intf/phys-[eth1/10]/dbgIfIn

标识与Ethernet1/10关联的输入统计信息对象。

理解DN的一种简单方法是将其视为DME层次结构中对象的完整地址。

传感器路径

传感器路径标识NX-OS为遥测订用必须监控的信息。

在初始实验示例中，DME可分辨名称用作传感器路径。后面的实际示例演示了预定义的资源路径标签。

例如：

```
sensor-group 2
path sys/intf/phys-[eth1/10]/dbgIfIn
path sys/intf/phys-[eth1/10]/dbgIfOut
path sys/intf/phys-[eth1/10]/phys
```

这些路径为Ethernet1/10提供输入统计信息、输出统计信息和操作信息。

数据编码

在NX-OS收集请求的信息后，必须先对数据进行编码，然后才能传输数据。

本实验中使用的编码是Google协议缓冲区(GPB)。

目标配置指定：

```
ip address 192.168.100.10 port 57000 protocol gRPC encoding GPB
```

GPB定义了如何在消息中表示收集的遥测信息。

编码和传输是独立的功能：GPB定义数据表示，而传输机制决定消息如何传送。

数据传输

本实验中使用的传输协议是gRPC。

Nexus交换机将GPB编码的遥测数据发送到：

192.168.100.10:57000

使用gRPC。

因此：

GPB定义了如何编码遥测信息。

gRPC提供传输机制，用于将遥测消息传送到接收器。

流遥测使用的gRPC传输不能与用于gRPC网络管理接口(gNMI)和gRPC网络操作接口(gNOI)等服务的NX-OS gRPC代理混淆。

遥测接收器

遥测接收器是接收和处理遥测流的外部系统或应用。

在本实验中，使用运行Telegraf的Ubuntu服务器作为接收器。接收器在TCP端口57000上侦听，并接受Nexus交换机生成的GPB-over-gRPC遥测流。

本实验中使用的接收器实施将在后面的“准备遥测接收器”部分中介绍。

目标组

目标组定义遥测数据必须发送到的位置以及传输方式。

本实验使用：

```
destination-group 1
ip address 192.168.100.10 port 57000 protocol gRPC encoding GPB
use-vrf default
```

定义用于遥测传输的接收器地址、目标端口、传输协议、编码以及虚拟路由和转发(VRF)实例。

传感器组

传感器组定义必须监控哪些信息。

例如：

```
sensor-group 2
path sys/intf/phys-[eth1/10]/dbgIfIn
path sys/intf/phys-[eth1/10]/dbgIfOut
path sys/intf/phys-[eth1/10]/phys
```

在本例中，传感器组2监控Ethernet1/10输入统计信息、输出统计信息和操作接口信息。dbgIfIn路径提供输入接口统计信息，dbgIfOut提供输出接口统计信息，phys提供接口的运行信息。

一个传感器组可以包含多个相关的传感器路径，因此可以回答以下问题：

收集哪些数据？

订用

订阅将传感器组与目标组关联并定义收集行为。

例如：

```
subscription 2
dst-grp 1
snsr-grp 2 sample-interval 60000
```

在本例中：

- 传感器组2确定收集的信息。
- 采样间隔确定收集行为。
- 目标组1确定信息发送的位置和方式。

因此，订用会连接主要遥测配置元素：

传感器组+收集行为+目标组

定期和基于事件的遥测

Cisco NX-OS流遥测支持基于DME的订用的定期和基于事件的收集。

收集行为由订阅内与传感器组关联的采样间隔控制。

定期遥感勘测

使用定期遥感勘测，NX-OS按配置的间隔收集和发送监控信息。

采样间隔以毫秒为单位指定。

For example: `snsr-grp 2 sample-interval 60000`

配置60秒的收集间隔。

定期遥测对于连续变化且通常随时间推移而分析的信息非常有用，例如：

- 接口数据包计数器
- 字节计数器
- 错误和丢弃
- 运营统计数据
- 资源利用率

在本实验中，订用1和订用2使用定期收集。

订用1每10秒收集一次Ethernet1/10接口对象，而订用2每60秒收集一次接口统计信息和操作信息。

基于事件的遥测

对于基于事件的遥测，NX-OS不使用定期收集计时器。

对于基于DME的遥测，基于事件的行为配置为：

`sample-interval 0`

当受监控对象更改时，遥测可生成与该更改关联的更新。

此收集方法对于以下信息很有用：

- 配置更改
- 管理状态更改
- 对象创建或删除
- 受监控DME对象下的其他更改

在本实验中，订用3监控Loopback100 DME对象：

```
sys/intf/lb-[lo100]:
snsr-grp 3 sample-interval 0
```

因此，本文档后面部分将使用对受监控Loopback100对象的更改来演示基于事件的遥测。

实验中使用的收集行为

初始实验配置中使用的三个订用可以总结如下：

订用	监控信息	采样间隔	收集行为
1	Ethernet1/10接口对象	10000 ms	定期
2	Ethernet1/10统计信息和运行状态	60000 ms	定期
3	Loopback100对象	0	基于事件的

主要区别在于用于生成遥测数据的触发器：

定期遥感勘测	基于事件的遥测
使用配置的计时器。	不使用定期计时器。
sample-interval > 0	sample-interval 0
生成重复样本。	当受监控对象发生更改时生成更新。
常用于计数器和统计信息。	常用于配置或状态更改。

配置和验证部分使用上述定义的订用演示这两种收集行为。

准备遥测接收器

流遥测需要能够接收和处理Nexus交换机发送的遥测数据的外部接收器。

对于本文档中使用的GPB-over-gRPC传输，接收方必须能够：

- 接收gRPC连接
- 在已配置的目标端口上侦听
- 解码GPB遥测消息
- 处理或存储接收的数据

遥测接收器的实施与本文档中介绍的NX-OS遥测配置无关。



注意：第三方遥测接收器软件的安装、配置、运行和故障排除均不在本文档的讨论范围之内。有关配置和支持信息，请参阅接收方供应商提供的文档。

本实验中使用的接收器

出于演示目的，本实验使用运行Telegraf的外部Ubuntu服务器作为遥测接收器。

接收器的参数为：

参数	价值
接收方地址	192.168.100.10
传输	gRPC
侦听端口	TCP/57000
编码	GPB

本文档未涵盖接收方软件配置。



注意：本文档中显示的接收器输出示例经过过滤，以突出显示与每个验证步骤相关的字段。

在Nexus交换机上配置流遥测之前，请验证：

- Nexus交换机和接收器之间存在IP连接。
- 接收器应用程序正在运行。
- 已配置的目标端口正在侦听。
- 接收方支持配置的传输协议。
- 接收器支持配置的遥测编码。

在本实验中，Nexus交换机上的Ethernet1/10使用192.168.100.1/24，遥测接收器使用192.168.100.10/24。在对遥测特定行为进行故障排除之前，必须确认基本的第3层可达性。

一旦接收器可访问并准备在TCP端口57000上接受GPB-over-gRPC遥感勘测，即可应用NX-OS遥感勘测配置。

在NX-OS上配置流遥测

该配置使用三个主要组件：

- 目标组:定义遥测数据的发送位置及其传输方式。
- 传感器组：定义收集哪些信息。
- 订用：将传感器组与目标组关联并定义收集行为。

本实验使用以下遥测目标：

参数	价值
目的地	192.168.100.10:57000
传输	gRPC
编码	GPB
VRF	默认

初始实验配置使用三个订用：

订用	传感器	收集类型	采样间隔
1	Ethernet1/10接口对象	定期	10000 ms
2	Ethernet1/10统计信息和运行状态	定期	60000 ms
3	Loopback100对象	基于事件的	0

启用遥测功能

必须先全局启用流遥测。

```
N9K-TELEMETRY-SW1# configure terminal
N9K-TELEMETRY-SW1(config)# feature telemetry
```

进入遥测配置模式：

```
N9K-TELEMETRY-SW1(config)# telemetry
N9K-TELEMETRY-SW1(config-telemetry)#
```

剩余遥测配置在此模式下执行。

配置目标组

目标组标识外部遥测接收器并定义用于发送遥测数据的传输和编码。

配置:

```
N9K-TELEMETRY-SW1(config-telemetry)# destination-group 1
N9K-TELEMETRY-SW1(conf-tm-dest)# ip address 192.168.100.10 port 57000 protocol gRPC encoding GPB
N9K-TELEMETRY-SW1(conf-tm-dest)# use-vrf default
```

此配置定义：

- 目的地址：192.168.100.10
- 目的端口：57000
- 传输：gRPC
- 编码：GPB
- VRF:默认

使用默认VRF是因为遥测接收器可通过默认VRF中的Ethernet1/10到达。

为遥测传输选择的VRF必须提供到已配置接收器的IP可达性。

配置传感器组1

传感器组1监控Ethernet1/10 DME接口对象。

```
N9K-TELEMETRY-SW1(config-telemetry)# sensor-group 1
N9K-TELEMETRY-SW1(conf-tm-sensor)# path sys/intf/phys-[eth1/10]
```

传感器路径标识DME分层结构中的Ethernet1/10托管对象并提供与该对象关联的通用接口信息。

配置订用1

将传感器组1与目标组1关联：

```
N9K-TELEMETRY-SW1(config-telemetry)# subscription 1
N9K-TELEMETRY-SW1(conf-tm-sub)# dst-grp 1
N9K-TELEMETRY-SW1(conf-tm-sub)# snsr-grp 1 sample-interval 10000
```

采样间隔以毫秒为单位指定。

10000 ms = 10秒

因此，订用1每10秒定期收集Ethernet1/10 DME对象并将遥测数据发送到目标组1。

配置传感器组2

传感器组2收集Ethernet1/10的统计信息和运行信息。

配置:

```
N9K-TELEMETRY-SW1(config-telemetry)# sensor-group 2
N9K-TELEMETRY-SW1(conf-tm-sensor)# path sys/intf/phys-[eth1/10]/dbgIfIn
N9K-TELEMETRY-SW1(conf-tm-sensor)# path sys/intf/phys-[eth1/10]/dbgIfOut
N9K-TELEMETRY-SW1(conf-tm-sensor)# path sys/intf/phys-[eth1/10]/phys
```

传感器路径提供：

传感器路径	信息
sys/intf/phys-[eth1/10]/dbgIfIn	输入接口统计信息
sys/intf/phys-[eth1/10]/dbgIfOut	输出接口统计信息
sys/intf/phys-[eth1/10]/phys	操作接口信息

一个传感器组可以包含多个相关的传感器路径，允许订阅从同一受监控接口收集多种类别的信息。

配置订用2

将传感器组2与目标组1关联：

```
N9K-TELEMETRY-SW1(config-telemetry)# subscription 2
N9K-TELEMETRY-SW1(conf-tm-sub)# dst-grp 1
N9K-TELEMETRY-SW1(conf-tm-sub)# snsr-grp 2 sample-interval 60000
```

配置的采样间隔为：

60000 ms = 60秒

因此，订用2每60秒收集一次Ethernet1/10统计信息和运行信息。

此订用稍后将在文档中用于验证定期遥测行为。

配置环回接口

环回接口用于演示基于事件的遥测，无需额外的物理连接。

配置:

```
N9K-TELEMETRY-SW1(config)# interface loopback100
N9K-TELEMETRY-SW1(config-if)# ip address 10.255.255.100/32
```

相应的DME对象为：

```
sys/intf/lb-[1o100]
```

环回接口提供了一种在基于事件的遥测验证期间生成受控配置和管理状态更改的简单方法。

配置传感器组3

配置Loopback100 DME传感器路径：

```
N9K-TELEMETRY-SW1(config-telemetry)# sensor-group 3
N9K-TELEMETRY-SW1(conf-tm-sensor)# path sys/intf/lb-[1o100]
```

传感器组3监控Loopback100托管对象。

配置订用3

将传感器组3与目标组1关联并配置基于事件的收集：

```
N9K-TELEMETRY-SW1(config-telemetry)# subscription 3
N9K-TELEMETRY-SW1(conf-tm-sub)# dst-grp 1
N9K-TELEMETRY-SW1(conf-tm-sub)# snsr-grp 3 sample-interval 0
```

对于基于DME的订用，零的样本间隔用于配置基于事件的行为。

因此，受监控Loopback100对象下的更改可以生成遥测通知，而无需使用循环收集计时器。

此订用稍后用于演示说明和管理状态更改。

完成初始遥测配置

完成配置后，使用以下命令进行验证：

```
N9K-TELEMETRY-SW1# show running-config telemetry
feature telemetry
telemetry
  destination-group 1
    ip address 192.168.100.10 port 57000 protocol gRPC encoding GPB
    use-vrf default
  sensor-group 1
    path sys/intf/phys-[eth1/10]
  sensor-group 2
    path sys/intf/phys-[eth1/10]/dbgIfIn
    path sys/intf/phys-[eth1/10]/dbgIfOut
    path sys/intf/phys-[eth1/10]/phys
  sensor-group 3
    path sys/intf/lb-[lo100]
  subscription 1
    dst-grp 1
    snsr-grp 1 sample-interval 10000
  subscription 2
    dst-grp 1
    snsr-grp 2 sample-interval 60000
  subscription 3
    dst-grp 1
    snsr-grp 3 sample-interval 0
```

此时，交换机具有开始向已配置的接收器发送遥测数据所需的目标、传感器组和订用。

下一部分验证传输会话和Ethernet1/10订用生成的定期遥感勘测。

验证定期遥感勘测

应用遥测配置后，验证传输会话已建立，周期性传感器组处于活动状态，并且正在收集遥测数据并将其传送到接收方。

验证遥测传输

运行此指令：

```
N9K-TELEMETRY-SW1# show telemetry transport
Session Id  Dst Grp  IP Address      Port      Encoding      Transport  Status
-----
0           1         192.168.100.10  57000     GPB           gRPC      Connected
-----
Retry buffer Size:          10485760
Event Retry Messages (Bytes): 0
Timer Retry Messages (Bytes): 0
Total Retries sent:         0
Total Retries Dropped:      0
N9K-TELEMETRY-SW1#
```

Connected状态确认已建立到已配置的遥测接收器的gRPC传输会话。

相关的传输参数也与已配置的目标组匹配。

检验周期性传感器组

使用:

```
N9K-TELEMETRY-SW1# show telemetry control database

Subscription Database size = 3
-----
Subscription ID      Data Collector Type  Reachability  Configuration Method
-----
3                     DME                  Reachable     CONFIG_DME-ADMIN
2                     DME                  Reachable     CONFIG_DME-ADMIN
```

Sensor Group Database size = 3

Row ID	Sensor Group ID	Sensor Group type	Sampling interval(ms)	Linked subscriptions	SubID
1	3	Event /DME	0/No Timer	1	3
Collection Time in ms (Cur/Min/Max): 0/0/0					
Encoding Time in ms (Cur/Min/Max): 0/0/0					
Transport Time in ms (Cur/Min/Max): 2/1/414					
Streaming Time in ms (Cur/Min/Max): 2/1/414					

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

Configuration method: CONFIG_DME-ADMIN

2	2	Timer /DME	60000/Running	1	2
Collection Time in ms (Cur/Min/Max): 0/0/1					
Encoding Time in ms (Cur/Min/Max): 0/0/1					
Transport Time in ms (Cur/Min/Max): 2/1/415					
Streaming Time in ms (Cur/Min/Max): 3/2/416					

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

Configuration method: CONFIG_DME-ADMIN

3	1	Timer /DME	10000/Running	1	1
Collection Time in ms (Cur/Min/Max): 0/0/1					
Encoding Time in ms (Cur/Min/Max): 0/0/1					
Transport Time in ms (Cur/Min/Max): 2/1/415					
Streaming Time in ms (Cur/Min/Max): 3/2/417					

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

Configuration method: CONFIG_DME-ADMIN

Row ID	Subscribed	Linked Groups	Sec Groups	Retrieve level	Path(GroupId) : Query : Filter
1	No	1	0	Self	sys/intf/phys-[eth1/10](1) : NA : NA
<snip>					
2	No	1	0	Self	sys/intf/phys-[eth1/10]/dbgIfIn(2) : NA : NA
<snip>					
4	No	1	0	Self	sys/intf/phys-[eth1/10]/dbgIfOut(2) : NA : NA

传感器组数据库确认周期性传感器组处于活动状态：

传感器组	类型	采样间隔	订用
1	计时器/DME	10000毫秒/运行	1

传感器组	类型	采样间隔	订用
2	计时器/DME	60000毫秒/运行2	

传感器组1每10秒收集一次Ethernet1/10接口对象。

传感器组2每60秒收集一次Ethernet1/10统计信息和运行信息。

同一命令还显示配置的传感器路径：

```
sys/intf/phys-[eth1/10]
sys/intf/phys-[eth1/10]/dbgIfIn
sys/intf/phys-[eth1/10]/dbgIfOut
sys/intf/phys-[eth1/10]/phys
```

对于本实验中使用的传感器路径，收集和编码时间通常介于0和1毫秒之间，并且未观察到遥测消息丢失。

验证数据收集

使用:

```
N9K-TELEMETRY-SW1# show telemetry data collector brief
```

Row ID	Collector Type	Successful	Payloads	Failed	Skipped
1	YANG	0	0	0	0
2	DME	513	513	0	44
3	NX-API	0	0	0	0

N9K-TELEMETRY-SW1#

实验报告如下：

```
DME Successful Collections: 513 Payloads: 513 Failed: 0 Skipped: 44 Dropped: 0
```

Successful和Payload计数器确认正在收集DME遥测数据，并且正在生成遥测负载。

44个跳过的收集是实验期间遥测目标暂时不可用时观察到的历史计数器。这些计数器稍后将在“基本

故障排除”部分中讨论。

有关每个传感器路径的其他信息，请使用：

```
N9K-TELEMETRY-SW1# show telemetry data collector details
```

此命令可以识别哪些配置的传感器路径有助于成功收集、失败收集、跳过收集或丢弃收集。

在接收方检验定期数据

订用2每60秒收集一次Ethernet1/10统计信息。

遥测接收器在21:44:45协调世界时(UTC)报告了以下输入统计信息：

```
timestamp: 2026-09-17T21:44:45Z
source: N9K-TELEMETRY-SW1
subscription: 2
path: sys/intf/phys-[eth1/10]/dbgIfIn
```

```
broadcastPkts: 5074
discards: 0
errors: 0
multicastPkts: 143403
octets: 12870535
ucastPkts: 31256
```

一分钟后，又收到另一个示例：

```
timestamp: 2026-09-17T21:45:45Z
source: N9K-TELEMETRY-SW1
subscription: 2
path: sys/intf/phys-[eth1/10]/dbgIfIn
```

```
broadcastPkts: 5075
discards: 0
errors: 0
multicastPkts: 143430
octets: 12875428
ucastPkts: 31292
```

两个样本之间的计数器更改汇总如下：

计数器	21:44:45	21:45:45
单播数据包	31,256	31,292
组播数据包	143,403	143,430
广播数据包	5,074	5,075
二进制八位数	12,870,535	12,875,428
错误	0	0
丢弃	0	0

时间戳之间大约相隔60秒，与为订用2配置的采样间隔匹配。

不断增加的数据包和字节计数器也确认正在收集更新的接口统计信息并将其传送给接收方。

检验输出和运行信息

dbgIfOut传感器路径提供Ethernet1/10的输出统计信息。

UTC时间21:45:45采集的样本报告称：

```
broadcastPkts: 8
discards: 0
errors: 0
multicastPkts: 4712
octets: 2444721
ucastPkts: 4092
```

phys传感器路径为同一接口提供操作属性。

接收方报告：

```
adminSt: up
operSt: up
operSpeed: 1G
operDuplex: full
operMtu: 1500
operDescr: TELEMETRY-COLLECTOR
```

这些示例确认，传感器组2通过其三个已配置的DME传感器路径提供接口统计信息和操作信息。

定期遥测验证摘要

定期遥测验证确认：

确认	结果
gRPC传输会话	已连接
GPB编码	已确认
传感器组1	运行时间：10000毫秒
传感器组2	运行时间：60000毫秒
DME集合	成功
失败的集合	0
丢弃的负载	0
周期性接收器样本	已接收
接口计数器	在样本之间更新

验证基于事件的遥测

验证定期遥测后，使用订用3通过生成对环回100托管对象的受控更改来验证基于事件的遥测。

验证基于事件的订用

使用:

N9K-TELEMETRY-SW1# show telemetry control database

Subscription Database size = 3

Subscription ID	Data Collector Type	Reachability	Configuration Method
3	DME	Reachable	CONFIG_DME-ADMIN
2	DME	Reachable	CONFIG_DME-ADMIN
1	DME	Reachable	CONFIG_DME-ADMIN

<snip>

Sensor Group Database size = 3

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

13Event /DME0/No Timer13

<snip>

Sensor Path Database size = 5

Row ID	Subscribed	Linked Groups	Sec Groups	Retrieve level	Path(GroupId) : Query : Filter
4	Yes	1	0	Self	sys/intf/lb-[lo100](3) : NA : NA

<snip>

Subscription Id: 3

Snapshot Stats:

Sent= 1Error= 0Drops= 0

The Sensor Group Database reports:

Sensor Group ID: 3Sensor Group Type: Event / DMESampling Interval: 0 / No TimerSubscription ID: 3

The Event / DME type and 0 / No Timer state confirm that Sensor Group 3 is configured for event-based c
The associated sensor path is:
sys/intf/lb-[lo100]

同一遥测数据库确认已通过订用3订用传感器路径。

生成基于事件的更新

要验证事件生成，请修改Loopback100说明：

```
N9K-TELEMETRY-SW1(config)# interface loopback100
N9K-TELEMETRY-SW1(config-if)# description TELEMETRY-EVENT-DEMO
```

遥测接收器报告：

```
{
"timestamp": "2026-09-17T21:41:12Z",
"source": "N9K-TELEMETRY-SW1",
"subscription": "3",
"event": {
"descr": "TELEMETRY-EVENT-DEMO",
"dn": "sys/intf/lb-[lo100]",
```

```
"status": "modified"
}
}
```

dn标识监控的DME对象，descr属性标识更改的属性。

接下来，管理性禁用并恢复接口：

```
N9K-TELEMETRY-SW1(config)# interface loopback100
N9K-TELEMETRY-SW1(config-if)# shutdown
```

然后：

```
N9K-TELEMETRY-SW1(config-if)# no shutdown
```

接收方检测到两个状态变化。

时间戳	已更改属性	价值
21:41:12	descr	TELEMETRY-EVENT-DEMO
21:41:19	adminSt	关闭
21:41:24	adminSt	up

每个更新都引用了相同的受监控对象：

```
sys/intf/lb-[lo100]
```

它报告对象状态为已修改。

这些结果证实，对受监控的受管对象不同属性的更改可以生成单独的遥测更新。

验证初始快照和事件消息

建立基于事件的订用时，NX-OS生成受监控对象的初始快照。

传感器路径数据库报告：

快照统计信息：

Sent = 1 Error = 0 Drops = 0

在三次受控更改后，报告同一传感器路径：

消息统计信息：

Sent = 3 Error = 0 Drops = 0

因此，结果可以总结为：

收集	计数
初始快照	1
说明修改	1
管理状态关闭	1
管理状态打开	1
总数	4

初始快照表示当订阅变为活动状态而后续消息对应于对象更改时受监视对象的状态。

验证事件收集统计信息

```
N9K-TELEMETRY-SW1# show telemetry event collector stats
-----
Row ID      Collection Count  Latest Collection Time      Sensor Path(GroupId)
-----
1           4                Thu Sep 17 21:41:24.045 UTC  sys/intf/1b-[1o100](3)
N9K-TELEMETRY-SW1#
```

实验报告如下：

Collection Count: 4
Latest Collection Time: Thu Sep 17 21:41:24.045 UTC
Sensor Path: sys/intf/1b-[1o100]

收集计数与一个初始快照以及测试期间生成的三个更改匹配。

验证事件收集器错误

N9K-TELEMETRY-SW1# show telemetry event collector errors

Error Description	Error Count
Dme Event Subscription Init Failures	- 0
Event Data Enqueue Failures	- 0
Event Subscription Failures	- 0
Pending Subscription List Create Failures	- 0
Subscription Hash Table Create Failures	- 0
Subscription Hash Table Destroy Failures	- 0
Subscription Hash Table Insert Failures	- 0
Subscription Hash Table Remove Failures	- 0
N9K-TELEMETRY-SW1#	

在测试期间未发现事件收集器错误。

基于事件的遥测验证摘要

基于事件的遥测验证确认：

确认	结果
传感器组	3
传感器路径	sys/intf/lb-[lo100]
收集类型	活动/DME
采样间隔	0 /无计时器
初始快照	已发送
说明更改	已检测
adminSt关闭	已检测
管理员启动	已检测
总收款	4
事件收集器错误	0

成功检测受控环回100更改可确认订用3按预期运行。

下一节将检查用于评估流遥测操作的主要验证和故障排除命令。

基本故障排除

当遥测数据未按预期接收时，请首先确定问题是否与传输会话、数据收集、传感器配置或外部接收器有关，然后再进行故障排除。

此工作流程使用本实验期间发现的问题，其中NX-OS报告了44个跳过收集和一个历史gRPC传输错误。

验证当前传输状态

```
N9K-TELEMETRY-SW1# show telemetry transport
Session Id  Dst Grp  IP Address      Port      Encoding      Transport  Status
-----
0           1        192.168.100.10  57000     GPB           gRPC       Connected

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

N9K-TELEMETRY-SW1#
```

在最终验证期间，遥测会话报告：

```
Session ID: 0
Destination Group: 1
IP Address: 192.168.100.10
Port: 57000
Encoding: GPB
Transport: gRPC
Status: Connected
```

“已连接”状态确认当前已建立传输会话。

但是，当前会话状态并不一定表示连接问题是否较早发生。因此，还应查看历史收集和传输计数器。

检查数据收集计数器

```
N9K-TELEMETRY-SW1# show telemetry data collector brief
-----
```

Row ID	Collector Type	Successful	Payloads	Failed	Skipped
1	YANG	0	0	0	0
2	DME	513	513	0	44
3	NX-API	0	0	0	0

N9K-TELEMETRY-SW1#

实验报告如下：

DME Successful Collections: 513
Payloads: 513
Failed: 0
Skipped: 44
Dropped: 0

Successful和Payload计数器确认已发生DME遥测收集和负载生成。

但是，“已跳过”计数器指示未执行44个计划收集。

要确定哪些传感器路径受到了影响，请使用：

N9K-TELEMETRY-SW1# show telemetry data collector details

Row ID	Successful	Payloads	Failed	Skipped	Dropped	Sensor Path(G
1	414	414	0	29	0	sys/intf/phys
2	54	54	0	5	0	sys/intf/phys
3	54	54	0	5	0	sys/intf/phys
4	54	54	0	5	0	sys/intf/phys

N9K-TELEMETRY-SW1#

报告的详细输出如下：

传感器路径	已跳过
sys/intf/phys-[eth1/10]	29
sys/intf/phys-[eth1/10]/dbglfIn	5
sys/intf/phys-[eth1/10]/dbglfOut	5
sys/intf/phys-[eth1/10]/phys	5
总数	44

跳过的集合分布于周期性传感器路径中，表明问题未隔离到单个DME对象。



注意：收集计数器是累积的。非零历史计数器并不一定表示当前存在相同的条件。

确定跳过收集的原因

```
N9K-TELEMETRY-SW1# show telemetry control stats
-----
Error Description                                     Error Count
-----
<snip>
Collections skipped due to destination unreachable      44
<snip>
N9K-TELEMETRY-SW1#
```

此输出直接标识了跳过收集的原因：

目的地不可达= 44

该值与DME数据收集器报告的44个跳过的收集匹配。

这使得调查可以远离传感器路径本身，转向遥测目标和传输路径。

检查传输错误

使用show telemetry transport报告的会话ID:

```
N9K-TELEMETRY-SW1# show telemetry transport 0 errors
Session Id:                                0
Connection Errors
  Connection Error Count:                   0
Transmission Errors
  Tx Error Count:                           1
  Last Tx Error:                            Thu Sep 17 21:22:16.008 UTC
  Last Tx Return Code:                      UNAVAILABLE
N9K-TELEMETRY-SW1#
```

实验报告如下：

Connection Error Count: 0

Tx Error Count: 1
Last Tx Error: Thu Sep 17 21:22:16.008 UTC
Last Tx Return Code: UNAVAILABLE

UNAVAILABLE返回代码记录与遥测目标关联的gRPC传输故障。

在本实验中，外部接收器是在修改配置时故意停止并重新启动的。在该间隔期间，NX-OS无法到达遥测目标，这与上面观察到的目标不可达收集计数器相对应。

重要的相关性在于：

观察	结果
已跳过收集	44
目标不可达	44
传输传输错误	1
最后发送返回代码	不可用
当前传输状态	已连接

已跳过匹配和无法到达目标的计数器提供了直接证据，说明已跳过收集的原因。

历史传输错误提供有关在同一实验练习中观察到的传输故障的其他信息。

验证传输恢复

恢复与接收方的连接后，使用：

```
N9K-TELEMETRY-SW1# show telemetry transport 0 stats
Session Id:                0
Connection Stats
  Connection Count          3
  Last Connected:           Thu Sep 17 21:27:16.010 UTC
  Disconnect Count          0
  Last Disconnected:        Never
Transmission Stats
  Compression:              disabled
  Source Interface:         not set()
  Transmit Count:           603
  Last TX time:             Thu Sep 17 21:51:36.008 UTC
  Min Tx Time:              1 ms
  Max Tx Time:              414 ms
  Avg Tx Time:              6 ms
  Cur Tx Time:              1 ms

Flow Stats
```

```
Allowed Queued Msgs Size (bytes): 78643200
Current Queued Msgs Size (bytes): 0
Utilization (percent): 0
Max Queued Msgs Size (bytes): 3849
Total Queued Msgs Size (bytes): 865313
Current Msgs Held (# Msgs): 0
Total Msgs held (# Msgs): 604
Max Msgs Held (# Msgs): 5
Msgs Dropped (# Msgs): 0
Flow Control Apply Time (secs): 0
Flow Control Last Applied: Never
<snip>
N9K-TELEMETRY-SW1#
```

最终实验室验证报告如下：

```
Connection Count: 3
Disconnect Count: 0
Transmit Count: 603
Average Transmission Time: 6 ms
Current Transmission Time: 1 ms
Current Queued Messages Size: 0
Transport Utilization: 0%
Messages Dropped: 0
Flow Control Last Applied: Never
```

这些值显示传输会话已恢复并且正在主动传输遥测数据，没有排队或丢弃的消息。

最重要的当前状态指标包括：

指示器	结果
传输状态	已连接
当前队列	0
邮件已丢弃	0
流量控制	从未应用

在对遥测计数器进行故障排除时，这种区别非常重要：即使基本条件已解决，历史错误仍可保持可见。

检查配置和事件收集器错误

其他命令可用于验证NX-OS是否报告配置或事件处理问题。

```
N9K-TELEMETRY-SW1# show telemetry config errors
```

Row ID	Path	Sensor Group	Error

Transport Errors			

Destination group ID	Source Interface	Configured VRF	Correct VRF

N9K-TELEMETRY-SW1#			

对于基于事件的遥测，请使用：

N9K-TELEMETRY-SW1# show telemetry event collector errors

Error Description	Error Count

Dme Event Subscription Init Failures	- 0
Event Data Enqueue Failures	- 0
Event Subscription Failures	- 0
Pending Subscription List Create Failures	- 0
Subscription Hash Table Create Failures	- 0
Subscription Hash Table Destroy Failures	- 0
Subscription Hash Table Insert Failures	- 0
Subscription Hash Table Remove Failures	- 0
N9K-TELEMETRY-SW1#	

所有事件收集器错误计数器也为零。

这些结果有助于在调查跳过的定期收集时排除配置和事件收集器故障。

故障排除工作流程

调查期间使用的命令可总结如下：

命令	目的
show telemetry transport	检查当前的传输会话状态。
show telemetry data collector brief	识别成功、失败、跳过或丢弃的集合。
显示遥测数据收集器详细信息	确定哪些传感器路径受到影响。
show telemetry control stats	确定跳过收集的原因。
show telemetry transport <session-id> errors	检查传输故障。
show telemetry transport <session-id> stats	检查传输恢复、队列和丢弃。
显示遥测配置错误	确定遥测配置错误。
show telemetry event collector errors	确定事件收集器错误。

在本实验中，故障排除序列确定了一个临时遥测目标可达性条件，而不是DME传感器路径或配置故障。

接收器再次可用后，遥测传输返回到“已连接”状态，收集恢复，并且未观察到当前队列或消息丢弃情况。

实际应用示例：监控系统资源

系统资源监控是流遥测的常见使用案例。CPU和内存信息可以定期从Nexus交换机导出到外部监控平台，用于历史分析、控制面板、容量监控和警报。

Cisco NX-OS为常用监控信息提供预定义的遥测路径标签。在本示例中，资源路径标签用于收集系统CPU和内存信息。

配置系统资源遥测

使用资源路径标签创建新的传感器组：

```
N9K-TELEMETRY-SW1(config)# telemetry
N9K-TELEMETRY-SW1(config-telemetry)# sensor-group 4
N9K-TELEMETRY-SW1(conf-tm-sensor)# path resources
```

创建订用4并将传感器组4与目标组1关联：

```
N9K-TELEMETRY-SW1(config-telemetry)# subscription 4
N9K-TELEMETRY-SW1(conf-tm-sub)# dst-grp 1
N9K-TELEMETRY-SW1(conf-tm-sub)# snsr-grp 4 sample-interval 10000
```

传感器组4使用预定义的资源路径标签。订用4将传感器组与现有遥测目标关联，并使用非零采样间隔配置定期收集。

相关配置为：

```
telemetry
 destination-group 1
   ip address 192.168.100.10 port 57000 protocol gRPC encoding GPB
   use-vrf default

 sensor-group 4
   path resources
```

```
subscription 4
  dst-grp 1
  snsrr-grp 4 sample-interval 10000
```

查看资源路径标签

运行此命令以检查由资源路径标签表示的DME路径：

```
N9K-TELEMETRY-SW1# show telemetry usability resources
```

```
1) label_name      : resources
   path_name       : sys/proc
   query_type      : poll
   <snip>

2) label_name      : resources
   path_name       : sys/procsys
   query_type      : poll
   <snip>

3) label_name      : resources
   path_name       : sys/procsys/systemem
   query_type      : event
   query_condition : query-target-filter=and(updated(procSysMem.memstatus),ne(procSysMem.memstatu
```

输出显示，资源标签表示多个底层DME路径。

sys/proc和sys/procsys路径使用轮询查询并提供进程和系统资源信息。sys/procsys/systemem路径使用事件查询，该查询可以在受监控内存状态更新且不再正常时报告更改。

这说明了直接指定单个DME可分辨名称和使用预定义遥测路径标签之间的区别。

例如：

```
Individual DME path:
path sys/intf/phys-[eth1/10]
Predefined path label:
path resources
```

验证系统资源遥测

使用show telemetry control database验证传感器组4和订用4的状态。

传感器组数据库报告：

Sensor Group ID	Sensor Group type	Sampling interval(ms)	Linked subscriptions	SubID
4	Timer /DME	10000/Running	1	4

计时器/DME类型和10000/运行采样间隔确认传感器组4作为周期性DME遥测源运行。

传感器路径数据库还显示与资源标签关联的基础路径。

例如：

```
resources:sys/procsys(4)
GPB Encoded Data size in bytes (Cur/Min/Max): 20221/20221/20229
Subscription Id: 4
Message Stats:
Sent = 14
Error = 0
Drops = 0
```

与流程相关的路径还报告活动的遥测收集：

```
resources:sys/proc(4)

GPB Encoded Data size in bytes (Cur/Min/Max): 82284/81262/85098
Subscription Id: 4

Message Stats:

Sent      = 14
Error     = 0
Drops     = 0
```

这些计数器确认正在收集资源信息，将其编码为GPB，并在传输时没有消息错误或丢弃。

请比较NX-OS系统资源

传统NX-OS CLI可用于显示交换机的当前CPU和内存状态：

```
N9K-TELEMETRY-SW1# show system resources
Load average:  1 minute: 0.57   5 minutes: 0.58   15 minutes: 0.63
Processes   :   854 total, 2 running
CPU states  :   14.64% user,   3.50% kernel,   81.85% idle
<snip>
Memory usage: 24530808K total,   9315248K used,   15215560K free
Kernel buffers: 22104K Used
Kernel cached : 6255520K Used
Current memory status: OK
```

CLI提供系统资源的即时视图。

流遥测允许将相同类型的CPU和内存信息导出到外部接收器，以便随着时间的推移可以存储和分析多个样本。

CPU利用率可能会快速变化。因此，在不同时间收集样本时，CLI显示的CPU值和通过遥测接收的值可能不同。

在接收方验证资源数据

遥测接收器成功解码了与订用4关联的系统资源信息。

此示例显示来自一个遥测示例的CPU和内存信息：

```
{
  "timestamp": "2026-09-18T23:15:08Z",
  "source": "N9K-TELEMETRY-SW1",
  "cpu": {
    "user": 11,
    "kernel": 2,
    "idle": 85,
    "averageLast60Seconds": 7.099999904632568
  },
  "memory": {
    "status": "OK",
    "utilization": 38.18336486816406,
    "usedKB": 9366688,
    "freeKB": 15164120,
    "totalKB": 24530808
  }
}
```

接收方输出确认正在成功解码来自订用4的CPU和内存信息并使其可用于外部监控。

为了进行比较，NX-OS CLI报告大约9.3 GB的已用内存（总内存约为24.5 GB），当前内存状态为

正常。遥测示例报告相同的总内存值、约38%的内存使用率以及相同的OK内存状态。

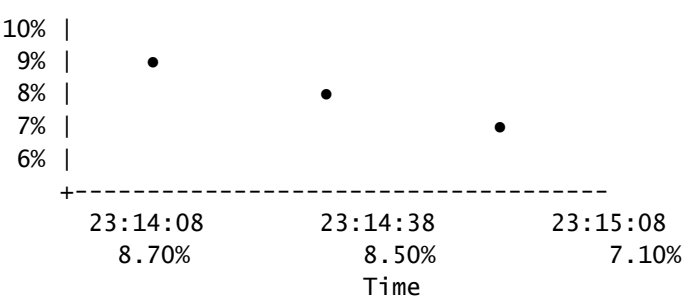
CPU值可能在CLI和遥测样本之间变化，因为CPU使用率会动态变化，而且不一定在同一时刻收集测量结果。

观察一段时间内的资源利用率

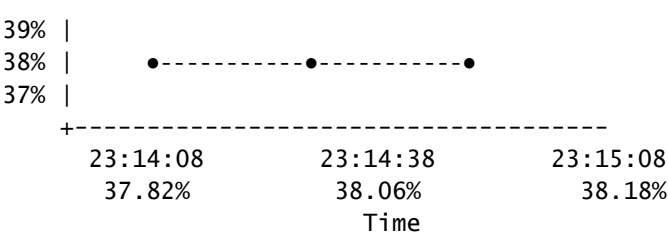
可以使用多个遥测样本来观察系统资源利用率随时间的变化。

在实验过程中，从订阅4收到了以下示例：

CPU平均利用率 — 最近60秒



内存利用率



这些示例说明了定期遥测如何提供基于时间的系统行为视图，而不是单个瞬时测量。在生产环境中，监控或可观察性平台可以存储大量的样本，并使用样本来识别趋势、生成警报和构建历史控制面板。

结论

Cisco NX-OS流遥测提供了一种结构化机制，用于将操作信息从Cisco Nexus 9000交换机导出到外

部遥测接收器。

本文档展示了流遥测的基本组件，包括DME传感器路径、GPB编码、gRPC传输、目标组、传感器组和订阅。

使用实验环境，配置和验证定期遥测和基于事件的遥测。定期订用用于收集Ethernet1/10统计信息 and 操作信息，而基于事件的订用用于检测对环回100托管对象的受控更改。

一个实际的系统资源监控示例还演示了如何使用预定义的 资源 路径标签将CPU和内存信息导出到外部遥测接收器。

验证和故障排除示例演示了NX-OS遥测命令如何用于验证传输连接、数据收集、事件处理和历史收集故障。

这些概念为了解、实施、验证和排除Cisco Nexus 9000 NX-OS上的基本流遥测部署提供了基础。

关于此翻译

思科采用人工翻译与机器翻译相结合的方式将此文档翻译成不同语言，希望全球的用户都能通过各自的语言得到支持性的内容。

请注意：即使是最好的机器翻译，其准确度也不及专业翻译人员的水平。

Cisco Systems, Inc. 对于翻译的准确性不承担任何责任，并建议您总是参考英文原始文档（已提供链接）。