Possibilities

#CiscoLive

Media Flow Analytics

Nexus 9000 Case Study

Varsha Dubey , Technical Consulting Engineer

DGTL-TSCDCN-303

Agenda

- Introduction
- The move to IP
- Operational Challenges
- Configuration and Verification of RTP flow monitoring on Nexus 9000
- Software Telemetry with Cisco Nexus 9000
- Conclusion

Introduction

- This session will cover about the move to IP and the associated challenges with monitoring IP Media Flows .
- We will look as to how Nexus 9000 can be leveraged to monitor IP Media Flows efficiently.
- We will go through configuration and verification of RTP flow monitoring on Nexus 9000.
- We will also look into configuring Software Telemetry on Nexus 9000.

The move to IP

IP based infrastructure

- Content production is moving from standard definition to high and ultra-high definition.

- Media companies are moving to IP-based infrastructures to meet the demands for more content and rich media experiences, including more camera feeds, higher resolutions, and virtual reality capabilities.

The move to IP

IP based infrastructure

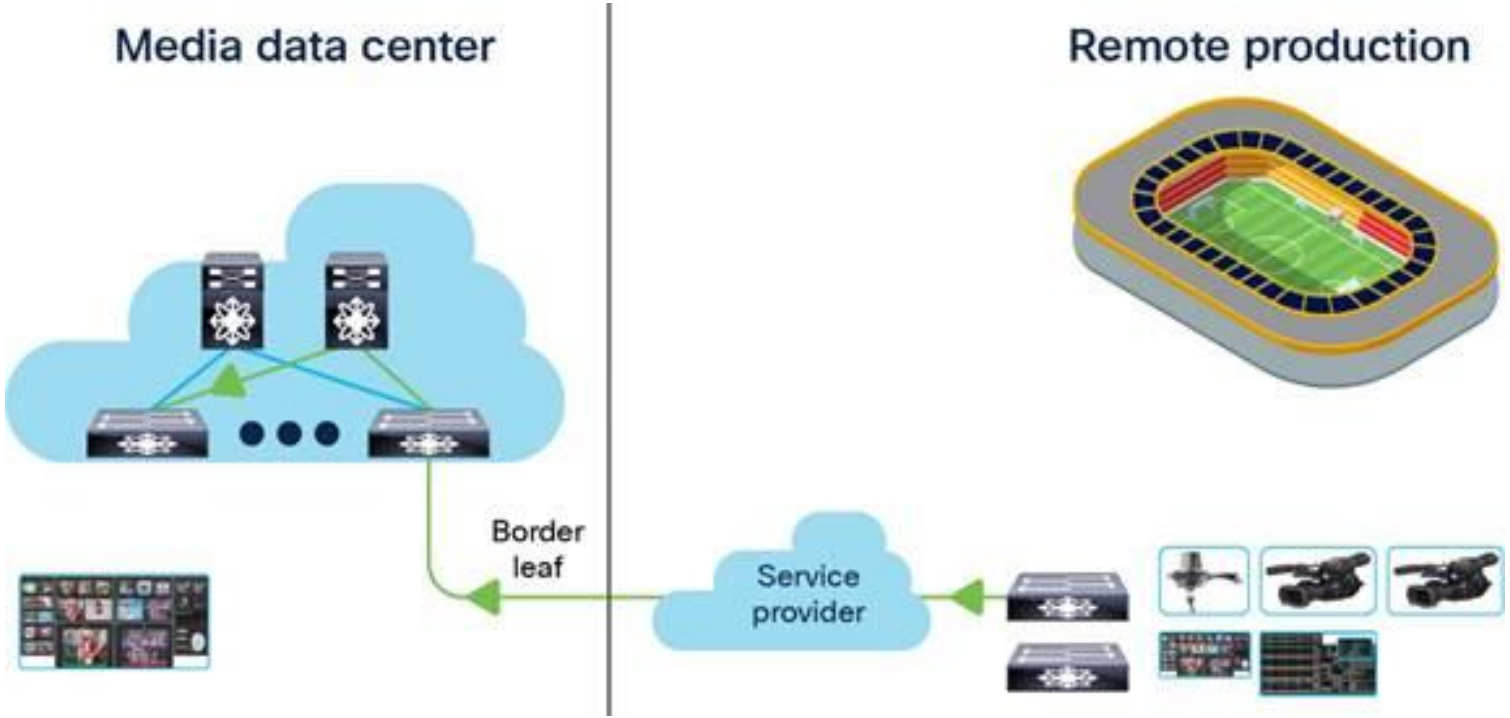
An IP network architecture makes production options such as live production in studios, stadiums, and remote production locations feasible.

This entails moving their production infrastructures from serial digital interfaces (SDI) to IP.

With an IP-based infrastructure, a single cable has the capacity to carry multiple bidirectional traffic flows and can support different flow sizes without requiring changes to the physical infrastructure.

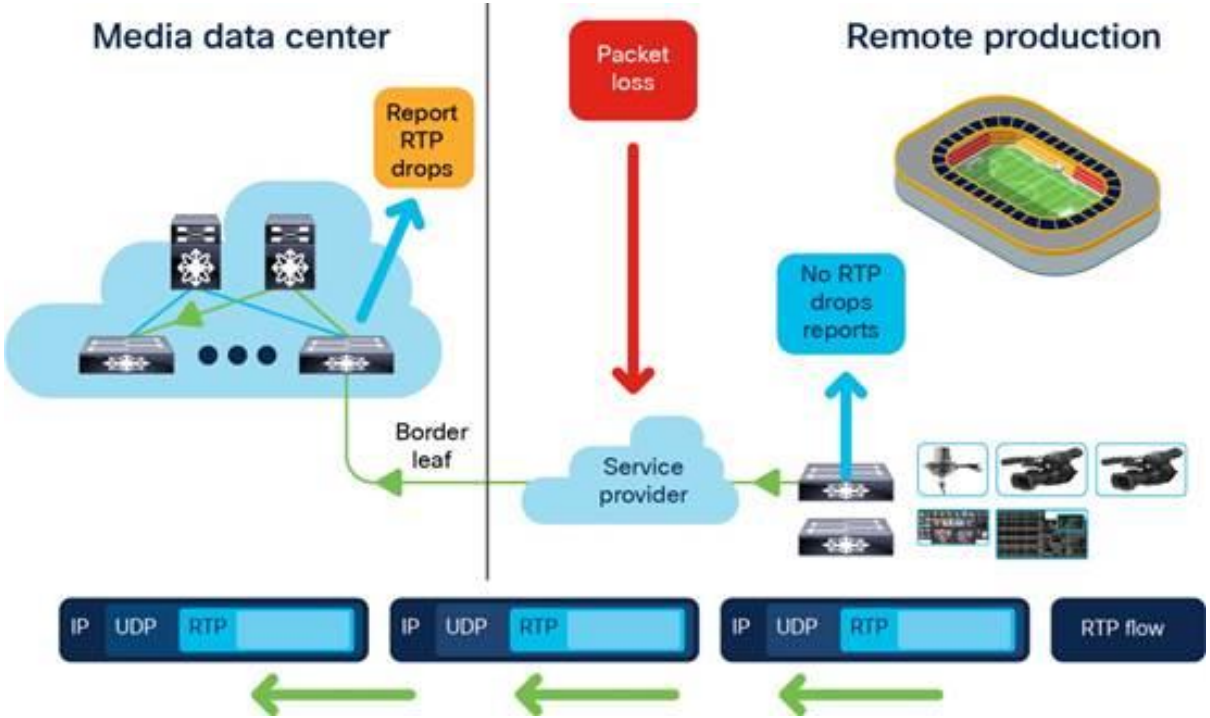
The move to IP

IP based infrastructure



Operational Challenges

IP based infrastructure



Operational Challenges

IP based infrastructure

- Bad video on air is an operator's nightmare.
- Interface errors and drops are beyond an operator's control.
- The lack of visibility into flow health in IP networks is a serious concern in the industry.
- If the video on screen goes bad because of packet loss, the loss could be happening at the stadium, at the service provider, or at the production facility itself.
- Media businesses need to be able to efficiently operate their IP networks and ensure reliability.

Solutions

IP based infrastructure

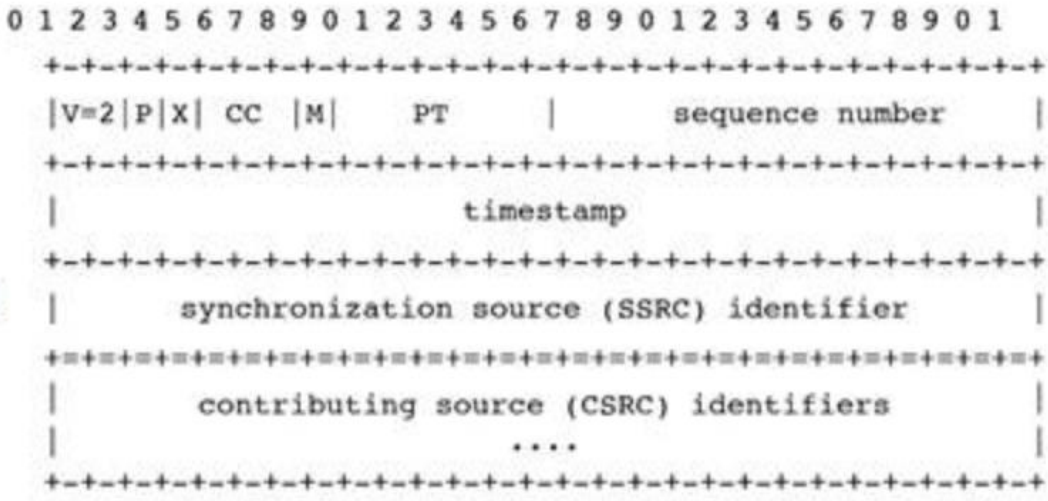
- **RTP flow monitoring on Nexus 9000**

- **Telemetry with the Cisco Nexus 9000 Series Switches and Cisco NX-OS**

RTP flow monitoring

RTP header

RTP headers include a sequence number which can be used to track loss



RTP flow monitoring

- RTP flow monitoring caches RTP flows on the switch and detects any gaps in the RTP sequence number, which indicates a loss in RTP frames.
- This information helps to pinpoint where the loss is occurring and enables you to better plan hardware resources.

Expected Sequence Number = (Last Sequence Number + Number of Packets) % (65536)

Number of Packets Lost = [65536 + (RTP Sequence Number – Expect Sequence Number)] % 65536

Guidelines and Limitations

Supported platforms are 9300-FX and 9300-FX2 .

NXOS release 9.3.1 and higher.

NetFlow and RTP flow monitoring cannot coexist on the switch.

You must reboot the switch after configuring UDF for RTP flow monitoring.

Media flow analytics uses the NetFlow TCAM region on the Cisco Nexus 9000 switch. The region is carved by default. If, for any reason, the region has not been carved, ensure that it is allocated before enabling the feature.

Configuring media flow analytics on Nexus 9000

Nexus_9k(config)# feature netflow

Nexus_9k(config)# udf netflow_rtp netflow-rtp

After this command a reboot is required

Nexus_9k(config)# ip|ipv6 flow rtp

This command creates a system wide ACL to filter the UDP port range of 16384 to 32767.

Nexus_9k (config)# ip access-list ipv4-test-acl

Nexus_9k (config)# 10 permit ip any 224.0.1.39/32 20 permit ip any 224.0.1.40/32

Configures an ACL policy to filter any specific traffic. This command is optional

Nexus_9k (config)# ip flow rtp ipv4-test-acl

Verifying media flow analytics on Nexus 9000

```
Nexus_9k# show running-config aclmgr
                ip access-list nfm-rtp-ipv4-acl
                   ignore routable
                   10 permit udp any any range 16384 32767
```

The ignore routable command filters any multicast traffic.

```
               ipv6 access-list nfm-rtp-ipv6-acl
                   ignore routable
                   10 permit udp any any range 16384 32767
```

Verifying media flow analytics on Nexus 9000

```
Nexus_9K# show system internal access-list input entries detail | grep 16384
```

```
[0x0001:0x0019:0x1219] permit udp 0.0.0.0/0 0.0.0.0/0 range 16384 32767   [0]
[0x0002:0x001a:0x121a] permit udp 0000:0000:0000:0000:0000:0000:0000:0000/0 00
00:0000:0000:0000:0000:0000:0000:0000/0 range 16384 32767 flow-label 262144  [0]
[0x0001:0x0019:0x1219] permit udp 0.0.0.0/0 0.0.0.0/0 range 16384 32767   [0]
[0x0002:0x001a:0x121a] permit udp 0000:0000:0000:0000:0000:0000:0000:0000/0 00
00:0000:0000:0000:0000:0000:0000:0000/0 range 16384 32767 flow-label 262144  [0]
```

Displaying RTP Flows and Errors

show flow rtp details

show flow rtp details {ipv4 | ipv6}

show flow rtp errors active

show flow rtp errors history

clear flow rtp detail

clear flow rtp detail {ipv4 | ipv6}

[no] flow rtp timeout *value*

Verifying RTP Flows and Errors

```
Nexus_9K # show flow rtp detail
RTP Flow timeout is 1440 minutes
IPV4 Entries
SIP       DIP       BD ID S-Port D-Port  Intf/Vlan Name  Packet Count  BytesPerSec  FlowStart
50.1.1.2 20.1.1.2 4151  16385  17999   Ethernet1/49/1  269207033     594468000    00:21:16 PST Jul 17 2020
20.1.1.2 50.1.1.2 4100  16385  18999   port-channel500  2844253      199000       00:21:59 PST Jul 17 2020


IPv6 Entries
SIP   DIP   BD ID S-Port D-Port  Intf/Vlan Name  Packet Count BytesPerSec  FlowStart
20::2 50::2 4100  30000  31999   port-channel500 2820074      199000       00:22:00 PST Jul 17 2020
50::2 20::2 4151  30000  31999   Ethernet1/49/1  3058232      199000       00:22:16 PST Jul 17 2020
```

Verifying RTP Flows and Errors

```
Nexus_9K # show flow rtp errors active
RTP Flow timeout is 1440 minutes
IPV4 Entries
SIP       DIP       BD ID  S-Port D-Port Intf/Vlan Name PacketCount BytesPerSec FlowStart
Packet Loss  Loss Start                Loss End

50.1.1.2  20.1.1.2  4151   16385  17999  Ethernet1/49/1  271757298   594469000   00:21:16 PST Jul 17 2020
56465837     00:21:59 PST Jul 17 2020  N/A

20.1.1.2 50.1.1.2   4100   16385  18999  port-channel500 2873224     199000      00:21:59 PST Jul 17 2020
302590234    00:22:15 PST Jul 17 2020  N/A
```

```
2020 Jul  17 00:21:59 Nexus_9K %NFM-1-RTP_FLOW_ERROR_DETECTED: Flow SIP: 50.1.1.2  DIP:
20.1.1.2 Interface: Ethernet1/49 loss detected
```

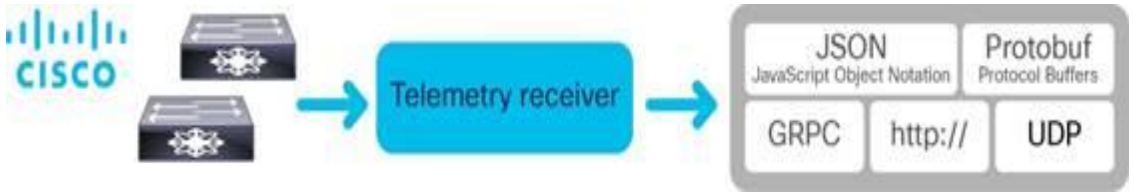
Verifying RTP Flows and Errors

```
Nexus_9K# show flow rtp errors history
RTP Flow timeout is 1440 minutes
IPV4 Entries
SIP       DIP       BD ID S-Port D-Port Intf/Vlan Name  Packet Count BytesPerSec FlowStart
Packet Loss Loss Start                Loss End
50.1.1.2  20.1.1.2  4151  16385  17999  Ethernet1/49/1  204187441    11122753    00:21:59 PST Jul 17 2020
2061        00:21:59 PST Jul 17 2020 00:22:15 PST Jul 17 2020
```

```
2020 Jul  17 00:22:15 Nexus_9K % NFM-1-RTP_FLOW_ERROR_STOP: Flow SIP: 50.1.1.2  DIP:
20.1.1.2 Interface: Ethernet1/49 loss no longer detected
```

Telemetry with the Cisco Nexus 9000 Series

Cisco NXOS



Rather than polling the network, the Cisco Nexus 9000 solution uses hardware and software telemetry to proactively notify operators of traffic problems.

Software Telemetry

Cisco Nexus 9000 and NXOS

- Cisco NX-OS provides several mechanisms such as SNMP, CLI, and Syslog to collect data from a network.
- These mechanisms have limitations that restrict automation and scale.
- One limitation is the use of the pull model, where the initial request for data from network elements originates from the client.
- A push model continuously streams data out of the network and notifies the client. Telemetry enables the push model, which provides near-real-time access to monitoring data.

Software Telemetry

- Components and Process

- Data collection
- Data Encoding
- Data Transport
- Telemetry Receiver

Data collection

DME and NXAPI

- Telemetry data is collected from the Data Management Engine (DME) database in branches of the object model specified using distinguished name (DN) paths.
- The data can be retrieved periodically (frequency-based) or only when a change occurs in any object on a specified path (event-based).
- NX-API is used to collect frequency-based data.

Encoding options

Google Protocol Buffers (GPB)

- Designed for simplicity, performance
- Highly efficient way of encoding telemetry data
- Not intended for human consumption

JavaScript Object Notation (JSON)

- Human-readable, self-describing, text-based encoding format
- Open-standard
- Not designed with performance or extensibility in mind

Transport Options

gRPC

- Open source RPC framework
- Low latency, scalable, distributed
- Enables extension such as authentication, load balancing, logging and monitoring and more

HTTP

- Omnipresent transport option
- Well known protocol
- Many available open source stacks on multiple operating systems

Telemetry Receiver

A telemetry receiver is a remote management system or application that stores the telemetry data.

The GPB encoder stores data in a generic key-value format. The encoder requires metadata in the form of a compiled .proto file to translate the data into GPB format.

In order to receive and decode the data stream correctly, the receiver requires the .proto file that describes the encoding and the transport services. The encoding decodes the binary stream into a key value string pair.

Software Streaming Telemetry Configuration

DME with GPB Encoding over gRPC Transport

```
feature telemetry
 telemetry
  destination-group 100
   ip address 1.2.3.4 port 50004 protocol gRPC encoding GPB
  sensor-group 100
    path sys/ch depth unbounded
    path sys/ospf depth unbounded
    subscription 600
  dst-grp 100
    snsr-grp 100 sample-interval 7000
```

Software Streaming Telemetry Configuration

NX-API with JSON Encoding over HTTP Transport

```
feature telemetry
 telemetry
  destination-group 1
    ip address 172.27.247.72 port 50102 protocol HTTP encoding JSON
  sensor-group 1
    path "show flow rtp details" depth 0
    path "show flow rtp errors active" depth 0
    subscription 1
  dst-grp 1
    snsr-grp 1 sample-interval 750000
```

Conclusion

Both Media Flow Monitoring and Software Telemetry on Nexus 9000 help to monitor IP Media flows efficiently.

RTP flow monitoring is easy to configure and does not require any external collector.

Software Telemetry requires an external collector and an elaborate configuration.

Thank you

Possibilities

#CiscoLive