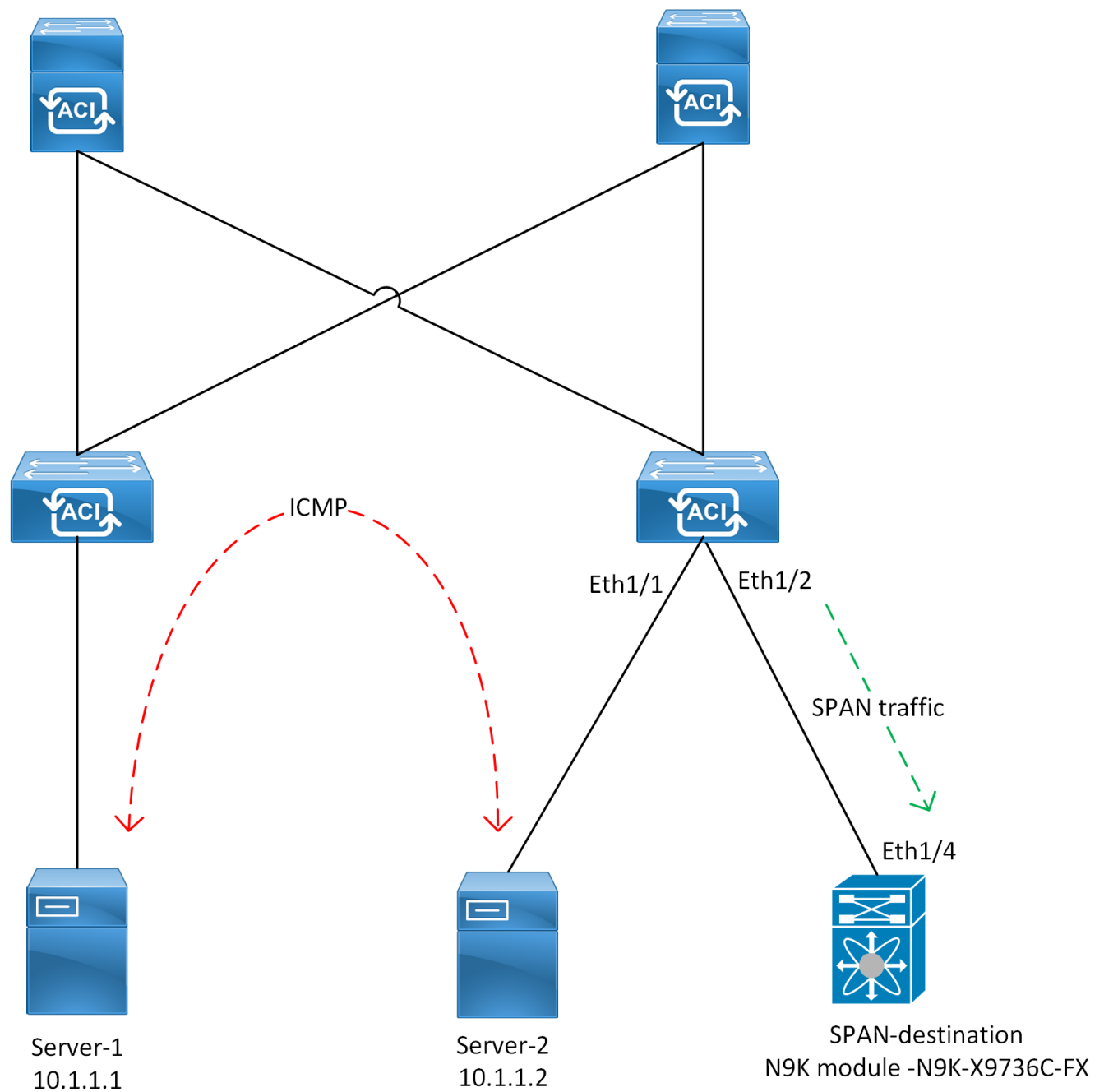


ACI Local SPAN (Access), Nexus 9000 Ethalyzer & SPAN-to-CPU



Lab by: Titus Majeza

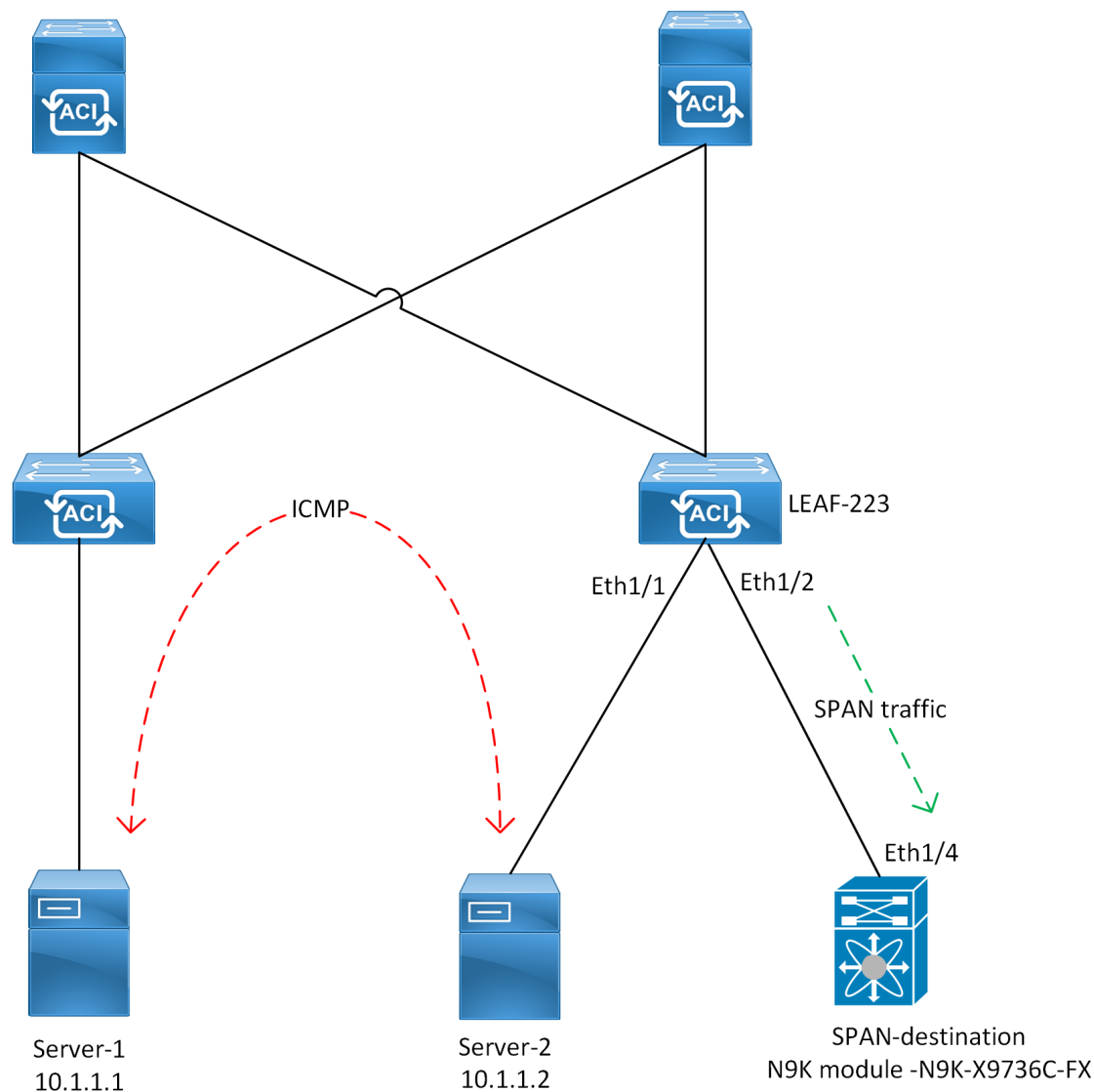


This lab configures a local Access Switch Port Analyzer (SPAN) session on Cisco ACI. To validate the SPAN functionality, a Cisco Nexus 9000 device is used as the SPAN destination (monitoring tool). The Nexus 9000's **SPAN-to-CPU** capability, combined with the **Ethalyzer** tool embedded in its operating system, enables it to serve as an effective monitoring tool.

Ethalyzer is an integrated packet analyzer in NX-OS, built on the command-line version of Wireshark. It can inspect packets that are either sent to the switch's supervisor or generated by the supervisor itself.

SPAN-to-CPU allows traffic from a specified interface on the Nexus switch to be redirected to its CPU interface. Once the traffic is punted to the CPU, Ethalyzer can be used to capture and analyse the packets of interest.

Lab Topology



The topology above is used for this lab exercise. Server-1 and Server-2 are in the same EPG and Server-2 is used as the source of the SPAN session. A Cisco Nexus 9000 switch is then used as the SPAN destination (i.e. monitoring station) and packets will be analysed from the device.

Pre-requisites

Before configuring ACI SPAN, ensure that the endpoints in ACI are properly learned within their respective EPGs and can communicate with each other.

In this lab, the servers (endpoints) are successfully learned in their assigned EPG.

EPG - EPG-Dev

SummaryPolicyOperational

Client EndpointsConfigured Access PoliciesContractsC

Healthy + - |

MAC/IP	Endpoint Name	Learning Source	Hosting Server	Reporting Interface (learned) Controller Name	Encap
✓ 54:7F:EE:13:77:BC 10.1.1.1		learned		Pod-1/Node-1002/eth1/31 (learned)	vlan-360
✓ 54:9F:C6:8F:16:3F 10.1.1.2		learned		Pod-1/Node-223/eth1/1 (learned)	vlan-360

SystemTenantsFabricVirtual NetworkingAdminOperationsAppsIntegrations

Visibility & Troubleshooting | Capacity Dashboard | EP Tracker | Visualization

EP Tracker

End Point Search

10.1.1.1

Learned At	Tenant	Application	EPG	IP
Pod:1, Leaf:1002, Port:eth1/31 (learned)	MIXED	AP_Mixed	EPG-Dev	10.1.1.1

Search

SystemTenantsFabricVirtual NetworkingAdminOperationsAppsIntegrations

Visibility & Troubleshooting | Capacity Dashboard | EP Tracker | Visualization

EP Tracker

End Point Search

10.1.1.2

Learned At	Tenant	Application	EPG	IP
Pod:1, Leaf:223, Port:eth1/1 (learned)	MIXED	AP_Mixed	EPG-Dev	10.1.1.2

Search

ACI SPAN Configuration & Verifications

SPAN copies traffic from the configured source port and sends the copied traffic to the configured destination (where a network analyzer is connected) for further analysis.

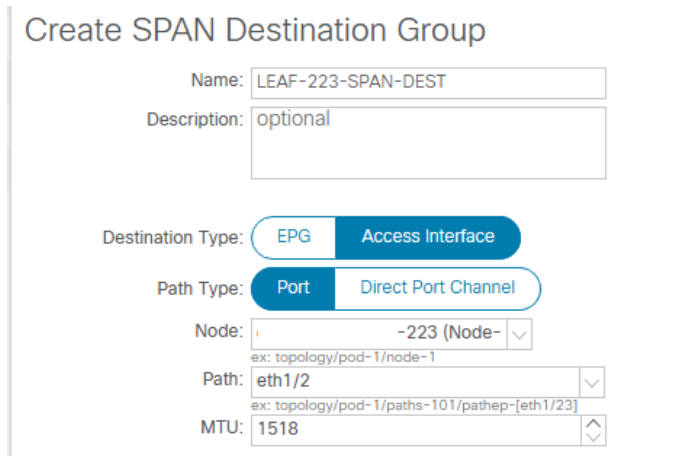
Info

For a local SPAN configuration, traffic on a source port is monitored and sent to a destination port local to the same leaf node.

SPAN is configured on ACI as follows.

Navigate to **Fabric >> Access Policies >> Policies >> Troubleshooting >> SPAN**

1. Configure the SPAN Destination Group.



Create SPAN Destination Group

Name: LEAF-223-SPAN-DEST

Description: optional

Destination Type: ☒ EPG ☐ Access Interface

Path Type: ☒ Port ☐ Direct Port Channel

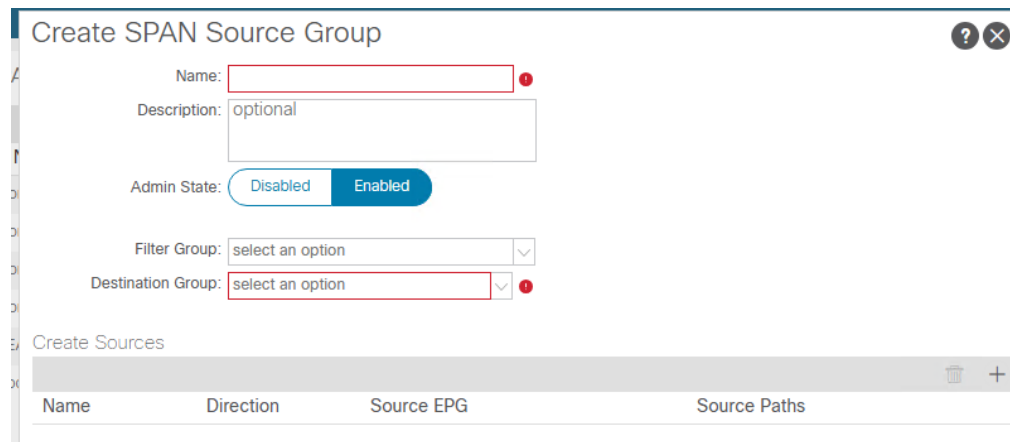
Node: -223 (Node-)
ex: topology/pod-1/node-1

Path: eth1/2
ex: topology/pod-1/paths-101/pathep-[eth1/23]

MTU: 1518

2. Configure the SPAN Source Group

The SPAN source group requires association with the Destination Group and the Source interface.



Create SPAN Source Group

Name:

Description: optional

Admin State: ☒ Disabled ☒ Enabled

Filter Group: select an option

Destination Group: select an option

Create Sources

Name	Direction	Source EPG	Source Paths
------	-----------	------------	--------------

The SPAN source is where the direction of desired traffic to be monitored is defined along with the source port whose traffic will be copied to the destination interface.

Create SPAN Source

A SPAN Source can either be configured for SPAN-on-drop or have a filter group associated to it, but not both. Note: If a source doesn't have a filter group assigned to it, it will receive a filter group from its source group (if it exists).

Name:

Description:

Direction: Both Incoming Outgoing

Filter Group:

Span Drop Packets: ☐

Type: None EPG Routed Outside

Add Source Access Paths

Source Access Path
Pod-1/Node-223/eth1/1

Associate Source to Path

Path Type: Port Direct Port Channel Virtual Port Channel VPC Component PC

Node:
ex: topology/pod-1/node-1

Path:
ex: topology/pod-1/paths-101/pathsep-[eth1/23]

The SPAN Source Group is configured with all its associated objects as follows:

Create SPAN Source Group

Name:

Description:

Admin State: Disabled Enabled

Filter Group:

Destination Group:

Create Sources

Name	Direction	Source EPG	Source Paths
SPAN-TX-RX	Both		Pod-1/Node-223/eth1/1

The desired operational status of the configured SPAN session can be observed:

SPAN Source Group - Local-SPAN-Access				
				Policy Operational
Node ID	Name	Session ID	Administrative State	Operational State
topology/pod-1/node-2...	infra_Local-SPAN-Access_LEAF-223-SPAN-DEST_LEAF-223-SPAN...	1	Enabled	up

The SPAN session is Enabled and its Operational State is “up”.

Nexus SPAN-to-CPU & Ethalyzer

As mentioned before, a Cisco Nexus 9000 will be used as a destination monitoring station. By default, SPAN replication is performed in hardware, and the supervisor CPU is not involved. To use the Ethalyzer capability, which only analyzes control plane packets, the SPAN-to-CPU capability is used. The SPAN-to-CPU functionality copies data plane traffic (in this case, SPAN packets) and sends it to the CPU. From this point, the Ethalyzer is then able to analyze traffic from the SPAN session.

Enable SPAN-to-CPU using the following configuration:

```
monitor session 1
  source interface ethernet1/4 both
  destination interface sup-eth0
  no shutdown
```

This configuration was used on the Nexus 9000 switch;

```
span-dest-N9K# show run monitor

!Command: show running-config monitor
!Time: Sat Feb  1 08:21:32 2025

version 7.0(3)I7(2)
monitor session 1
  source interface Ethernet1/4 both
  destination interface sup-eth0
  no shut

span-dest-N9K#
```

Interface **Ethernet1/4** is connected to the ACI leaf port that is receiving SPAN traffic. This interface (**Eth1/4**) remains in its default state with no configuration applied.

```
span-dest-N9K# show run interface ethernet1/4

!Command: show running-config interface Ethernet1/4
!Time: Sat Feb  1 11:51:10 2025

version 7.0(3)I7(2)

interface Ethernet1/4
  no shutdown

span-dest-N9K#
```

Verify the monitor session on the Nexus 9000 device:

```
span-dest-N9K# show monitor
Session  State          Reason                Description
-----  -
1         up                The session is up
span-dest-N9K#
```

```
span-dest-N9K#
span-dest-N9K# show monitor session 1
session 1
-----
type           : local
state          : up
acl-name       : acl-name not specified
source intf    :
  rx           : Eth1/4
  tx           : Eth1/4
  both         : Eth1/4
source VLANs   :
  rx           :
  tx           :
  both         :
filter VLANs   : filter not specified
source fwd drops :
destination ports : sup-eth0

span-dest-N9K#
```

The monitor session is in an “up” state.

Testing

To test if the Nexus 9000 switch is indeed receiving SPAN packets from ACI, Ethanalyzer is used.

1. Server-2 initiates ICMP traffic to Server-1.

```
server-2# ping 10.1.1.1
PING 10.1.1.1 (10.1.1.1): 56 data bytes
64 bytes from 10.1.1.1: icmp_seq=0 ttl=254 time=1.04 ms
64 bytes from 10.1.1.1: icmp_seq=1 ttl=254 time=0.77 ms
64 bytes from 10.1.1.1: icmp_seq=2 ttl=254 time=0.776 ms
64 bytes from 10.1.1.1: icmp_seq=3 ttl=254 time=13.202 ms
64 bytes from 10.1.1.1: icmp_seq=4 ttl=254 time=3.786 ms

--- 10.1.1.1 ping statistics ---
5 packets transmitted, 5 packets received, 0.00% packet loss
round-trip min/avg/max = 0.77/3.914/13.202 ms
server-2#
```

2. On the Nexus 9000 switch the ICMP packets are successfully captured for analysis:

```
span-dest-N9K# ethanalyzer local interface inband display-filter "icmp" limit-captured-frames 0
```

```
Capturing on inband
2025-02-01 08:24:28.064052    10.1.1.2 -> 10.1.1.1    ICMP Echo (ping) request
2025-02-01 08:24:28.064379    10.1.1.1 -> 10.1.1.2    ICMP Echo (ping) reply
2025-02-01 08:24:28.064987    10.1.1.2 -> 10.1.1.1    ICMP Echo (ping) request
2025-02-01 08:24:28.065228    10.1.1.1 -> 10.1.1.2    ICMP Echo (ping) reply
2025-02-01 08:24:28.065867    10.1.1.2 -> 10.1.1.1    ICMP Echo (ping) request
2025-02-01 08:24:28.066105    10.1.1.1 -> 10.1.1.2    ICMP Echo (ping) reply
2025-02-01 08:24:28.066723    10.1.1.2 -> 10.1.1.1    ICMP Echo (ping) request
2025-02-01 08:24:28.073443    10.1.1.1 -> 10.1.1.2    ICMP Echo (ping) reply
2025-02-01 08:24:28.074041    10.1.1.2 -> 10.1.1.1    ICMP Echo (ping) request
2025-02-01 08:24:28.083057    10.1.1.1 -> 10.1.1.2    ICMP Echo (ping) reply
```

```
10 packets captured
```

```
span-dest-N9K# ethanalyzer local interface inband mirror display-filter "icmp" limit-captured-frames 0 detail
```

```
Capturing on inband
Frame 32 (102 bytes on wire, 102 bytes captured)
  Arrival Time: Feb  1, 2025 09:17:46.048679000
  [Time delta from previous captured frame: 1.059201000 seconds]
  [Time delta from previous displayed frame: 5.059593000 seconds]
  [Time since reference or first frame: 5.059593000 seconds]
  Frame Number: 32
  Frame Length: 102 bytes
  Capture Length: 102 bytes
  [Frame is marked: False]
  [Protocols in frame: eth:vlan:ip:icmp:data]
Ethernet II, Src: 54:9f:c6:8f:16:3f (54:9f:c6:8f:16:3f), Dst: 54:7f:ee:13:77:bc (54:7f:ee:13:77:bc)
  Destination: 54:7f:ee:13:77:bc (54:7f:ee:13:77:bc)
    Address: 54:7f:ee:13:77:bc (54:7f:ee:13:77:bc)
      .... 0 .... = IG bit: Individual address (unicast)
      .... 0. .... = LG bit: Globally unique address (factory default)
  Source: 54:9f:c6:8f:16:3f (54:9f:c6:8f:16:3f)
    Address: 54:9f:c6:8f:16:3f (54:9f:c6:8f:16:3f)
      .... 0 .... = IG bit: Individual address (unicast)
      .... 0. .... = LG bit: Globally unique address (factory default)
  Type: 802.1Q Virtual LAN (0x8100)
802.1Q Virtual LAN, PRI: 0, CFI: 0, ID: 360
  000. .... = Priority: 0
  ...0 .... = CFI: 0
  .... 0001 0110 1000 = ID: 360
  Type: IP (0x0800)
Internet Protocol, Src: 10.1.1.2 (10.1.1.2), Dst: 10.1.1.1 (10.1.1.1)
  Version: 4
  Header length: 20 bytes
  Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
    0000 00.. = Differentiated Services Codepoint: Default (0x00)
    .... 0. = ECN-Capable Transport (ECT): 0
    .... 0 = ECN-CE: 0
  Total Length: 84
```

This lab successfully demonstrated the configuration and verification of a local SPAN session in Cisco ACI, with a Cisco Nexus 9000 switch serving as the SPAN destination. By leveraging the **SPAN-to-CPU** capability, traffic was redirected to the Nexus CPU interface and analysed using **Ethanalyzer**, a built-in packet capture tool in NX-OS.