

DC-HA

Overview:

In any SDWAN solutions, availability of DC is always more critical. As an interim solution, we support only ACTIVE-PASSIVE mode in DC. In this design, there will be only two DC's per tenant in the HA cluster and there will be a Gateway node through which both the DC's reach their LAN network.

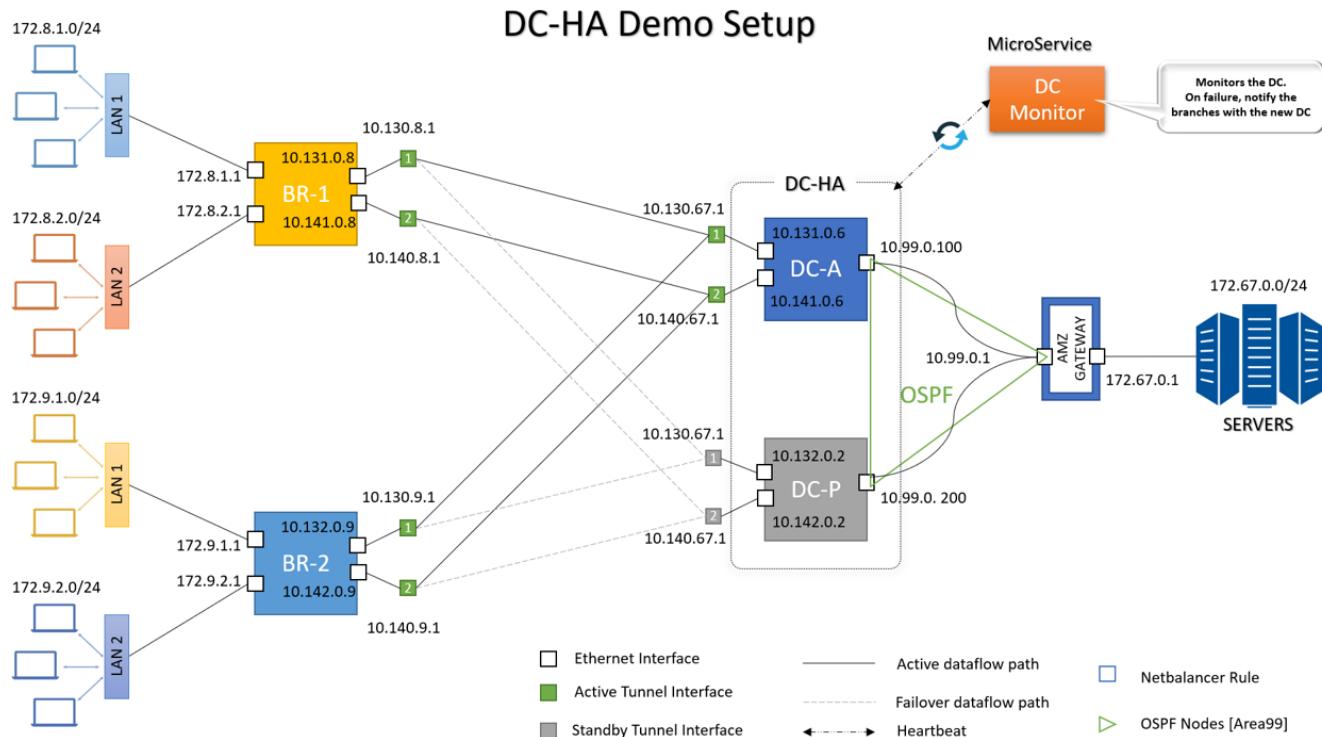
All the DC's will have public IP [WAN] as an endpoint through which all the branches will establish their IPSEC tunnels. And the tunnels are usually one to many from DC. So, if the ACTIVE DC fails that failure needs to be informed to all of their branches and reestablish the connection from the PASSIVE DC.

As of today, topology has the all the details about both the branches and their DCs w.r.t tunnels.

So, we have created a new micro-service named **dc-monitor**, that checks the liveness of ACTIVE DC and collects the topology for every few [say 20] seconds. If the **dc-monitor** detects any failure, then it checks for any passive DC is available to takeover. If that is available, then the **dc-monitor** informs all the branches to connect to the PASSIVE DC and change its state to new ACTIVE.

During these switchover, the LAN behind the DC's should reach the branches through new ACTIVE DC. Someone needs to inject the reverse route here, we are achieving this by having our vCPE as Gateway node that connects both the ACTIVE and PASSIVE DCs and all the DC-LAN traffic is reached through that Gateway. Also OSPF is configured between DC-ACTIVE, DC-PASSIVE and Gateway node, so that the reverse routed is injected seamlessly without any issues even during DC failover and failback.

Please see the overview of our setup as shown in the *figure* below:



Setup Configuration:

Make sure that your setup is configured as shown in the *figure* above. It has 5 main nodes:

- DC-A (HA_DC1 runs in ACTIVE mode)
- DC-P (HA_DC2 runs in Passive mode)
- AMZ-Gateway (AMZ_LAN_GW node behind DC nodes)
- BR-1 (Branch 1)
- BR-2 (Branch 2)

1. Onboard AMZ_LAN_GW, HA_DC1, HA_DC2, BR1 & BR2 CPE's from secure shell. After onboarded, the Alpsee UI should list like shown below:

NOTE: Make sure that **dc_monitor** micro-service is running in the onboarding provider, without that HA will not function.

- While onboarding, add the **metadata** to differentiate DC and BR CPE's.

For HA_DC1 --> CPE_TYPE = DC, CPE_MODE = HA, HA_STATE = ACTIVE
 For HA_DC2 --> CPE_TYPE = DC, CPE_MODE = HA, HA_STATE = PASSIVE

Edit Edge Controller

Product ID	ZKVMAZA00000001-68PKFP9FPCJ3R5VMIG7LNH35TC
Tunnel IP	10.11.255.12
Tunnel Port	1194
Name	DC-A
Description	Data Center A

Meta Data

Key	Value	+
longitude:-84.388	latitude:33.749	location:Atlanta
HA_STATE:ACTIVE	CPE_MODE:HA	CPE_TYPE:DC

[CANCEL](#) [UPDATE](#)

Edit Edge Controller

Product ID	ZKVMAZA00000001-5NI0IWT200K66ENRWOALR028MJ
Tunnel IP	10.11.255.12
Tunnel Port	1194
Name	DC-B
Description	Data Center B

Meta Data

Key	Value	+
longitude:-74.006	latitude:40.7128	location:Newyork
HA_STATE:PASSIVE	CPE_MODE:HA	CPE_TYPE:DC

[CANCEL](#) [UPDATE](#)

For both BR --> CPE_TYPE = BR

Edit Edge Controller

Product ID	ZKVMAZA00000001-4442BFZ9HLR6SA40ITLQDV19WV
Tunnel IP	10.11.255.12
Tunnel Port	1194
Name	BR-1
Description	Branch 1

Meta Data

Key	Value	+
CPE_TYPE:BR	longitude:80.2707	latitude:13.0827
location:Chennai		

[CANCEL](#) [UPDATE](#)

Edit Edge Controller

Product ID	ZKVMAZA00000001-BRM794KCBHSQVIU9ESCE420P2
Tunnel IP	10.11.255.12
Tunnel Port	1194
Name	BR-2
Description	Branch 2

Meta Data

Key	Value	+
CPE_TYPE:BR	longitude:77.1025	latitude:28.7041
location:Delhi		

[CANCEL](#) [UPDATE](#)

- After onboarding everything, **list** all the edge controllers

The screenshot shows the WAN Provider Device Management interface. On the left, there's a sidebar with sections like Device Management, Analytics, User Management, and Configuration. The main area is titled 'Device Management > Edge Controllers'. It has tabs for GROUPS, LIST, and SITE AVAILABILITY. A search bar is at the top right. Below is a table with columns: Name, Description, Location, and Interface Status. The table lists five devices:

Name	Description	Location	Interface Status
BR-2	Branch 2	N/A	IPSEC00 ↑ IPSEC01 ↑ LAN00 ↑ LAN01 ↑ WAN00 ↑ WAN01 ↑
BR-1	Branch 1	N/A	IPSEC00 ↑ IPSEC01 ↑ LAN00 ↑ LAN01 ↑ WAN00 ↑ WAN01 ↑
DC-B	Data Center B	N/A	IPSEC00 ↓ IPSEC01 ↓ LAN00 ↑ WAN00 ↑ WAN01 ↑
DC-A	Data Center A	N/A	IPSEC00 ↑ IPSEC01 ↑ LAN00 ↑ WAN00 ↑ WAN01 ↑
AMZ-LAN-GW	AMZ LAN Gateway	N/A	LAN00 ↑ LAN01 ↑ WAN00 ↑ WAN01 ↑

At the bottom, there are buttons for 'Rows per page' (set to 10) and navigation arrows.

2. Now configure the OSPF in AMZ_LAN_GW, HA_DC1 & HA_DC2 CPE's [for reverse route injection]

OSPF Configuration

Please Configure OSPF on AMZ_GATEWAY with the below mentioned steps:

- Enable OSPF

The screenshot shows a network configuration interface with a top navigation bar containing tabs for ROUTES, FILTER LISTS, ROUTE MAPS, OSPF, BGP, NAT, MULTICAST, and PORT FORWARDING. Below the navigation bar, there are several status indicators and buttons: a gear icon, a 'STATUS' button, and a 'ROUTE REDISTRIBUTION' button. A 'Router ID: 0.0.0.0' field is present with an 'Enabled' toggle switch set to 'Enabled'. Below this, there is a 'Enable / Disable' button and an 'ADD AREA' button. The main content area is titled 'Areas' and contains a table with columns: Area ID, Description, Type, and Auth Type. The table displays the message 'No Records Found'. At the bottom of the screen, a green notification box shows a checkmark icon and the text 'OSPF Config enabled successfully'.

2. Add the RouterID

OSPF Global Advanced Settings

Options

Router ID 9.9.9.9	Log Adjacency disabled (default)	SPF Delay 0
SPF Hold Time 50	SPF Max Hold Time 5000	Default: 0, Range: 0 - 600000 Default: 50, Range: 0 - 600000 Default: 5000, Range: 0 - 600000

Default Route Redistribution

<input type="checkbox"/> Redistribute Default	<input type="checkbox"/> Always Redistribute	
Metric 20	Metric Type 2	Route Map
Default: 20, Range: 0 - 16777214	Default: 2, Range: 1 - 2	

Advanced

<input type="checkbox"/> Opaque LSA	<input type="checkbox"/> RFC 1583 Compatibility	
<input type="checkbox"/> Max Metric	Startup Seconds 0	Shutdown Seconds 0
Reference Bandwidth 100000	Write Multiplier 20	ABR Type cisco (default)
Default: 100000, Range: 1 - 4294967	Default: 20, Range: 1 - 100	Default: 0 (Disable), Range: 5 - 100 Default: 0 (Disable), Range: 5 - 100

Distance Information

CANCEL **SAVE**

3. Add new area

Add Area

Area ID: 99

Description: amz_gw

IP Address or Range: 0 - 4294967295

Type: Normal

NSSA Translator Role:

Default Cost: 16777215

Shortcut: Default

Authentication

Auth Type: None

ABR Summary Route Filtering

Export List:

Import List:

Filter List Out:

Filter List In:

CANCEL **ADD**

4. Add active interface through which the routes are learned and select the network type as ***point_to_multicast***

Add Interface

Area ID 99	Description
Interface LAN01	Network Type point_to_multipoint
<input checked="" type="checkbox"/> Passive Interface	
Advanced Options	
Link Cost 10 Default: 10, Range: 1 - 65535	DR Priority 1 Default: 1, Range: 0 - 255
Retransmit Interval 5 Default: 5, Range: 1 - 65535	Transmit Delay 1 Default: 1, Range: 1 - 65535
Hello Interval 10 Default: 10, Range: 1 - 65535	Dead Interval 40 Default: 40, Range: 1 - 65535
<input type="checkbox"/> Hello Multiplier 0 Range: 1 - 10	<input checked="" type="checkbox"/> Ignore MTU
Authentication	
Auth Type None	Auth Key
Digest Key ID	Digest Key

CANCEL **ADD**

5. Add passive interface through which the DC LAN servers are reachable and select the network type as **broadcast**

Add Interface

Area ID — 99 Description

Interface — LAN00 Network Type — broadcast

Passive Interface

Advanced Options

Link Cost — 10 Default: 10, Range: 1 - 65535	DR Priority — 1 Default: 1, Range: 0 - 255
Retransmit Interval — 5 Default: 5, Range: 1 - 65535	Transmit Delay — 1 Default: 1, Range: 1 - 65535
Hello Interval — 10 Default: 10, Range: 1 - 65535	Dead Interval — 40 Default: 40, Range: 1 - 65535
<input type="checkbox"/> Hello Multiplier — 0 Range: 1 - 10	<input checked="" type="checkbox"/> Ignore MTU

Authentication

Auth Type — None Auth Key

Digest Key ID Digest Key

CANCEL **ADD**

Please Configure OSPF on DC-A with the below mentioned steps:

1. Enable OSPF

The screenshot shows a network configuration interface with a top navigation bar containing tabs for ROUTES, FILTER LISTS, ROUTE MAPS, OSPF, BGP, NAT, MULTICAST, and PORT FORWARDING. Below the navigation bar, there is a status bar with icons for STATUS and ROUTE REDISTRIBUTION. A main content area displays the following information:

- Router ID:** 0.0.0.0
- Enabled:** A toggle switch is set to Enabled.
- Enable / Disable:** A button to toggle the OSPF enablement.
- ADD AREA:** A button to add a new OSPF area.
- Areas:** A table header with columns: Area ID, Description, Type, and Auth Type. The table body displays the message: No Records Found.
- OSPF Config enabled successfully:** A green notification message with a checkmark icon.

2. Add the RouterID

OSPF Global Advanced Settings

Options

Router ID 1.1.1.1	Log Adjacency disabled (default)	SPF Delay 0
SPF Hold Time 50	SPF Max Hold Time 5000	Default: 0, Range: 0 - 600000 Default: 5000, Range: 0 - 600000

Default Route Redistribution

<input type="checkbox"/> Redistribute Default	<input type="checkbox"/> Always Redistribute	
Metric 20	Metric Type 2	Route Map
Default: 20, Range: 0 - 16777214	Default: 2, Range: 1 - 2	

Advanced

<input type="checkbox"/> Opaque LSA	<input type="checkbox"/> RFC 1583 Compatibility	
<input type="checkbox"/> Max Metric	Startup Seconds 0	Shutdown Seconds 0
Reference Bandwidth 100000	Write Multiplier 20	ABR Type cisco (default)
Default: 100000, Range: 1 - 4294967	Default: 20, Range: 1 - 100	Default: 0 (Disable), Range: 5 - 86400 Default: 0 (Disable), Range: 5 - 100

Distance Information

CANCEL SAVE

3. Add the redistribution for all kernel routes

Route Redistribution

<input type="checkbox"/> Connected	Metric Range: 0 - 16777214	Metric Type Default: 2, Range: 1 - 2	Route Map	Distribute List
<input checked="" type="checkbox"/> Kernel	Metric 0 Range: 0 - 16777214	Metric Type 2 Default: 2, Range: 1 - 2	Route Map	Distribute List
<input type="checkbox"/> BGP	Metric Range: 0 - 16777214	Metric Type Default: 2, Range: 1 - 2	Route Map	Distribute List
<input type="checkbox"/> Static	Metric Range: 0 - 16777214	Metric Type Default: 2, Range: 1 - 2	Route Map	Distribute List

CANCEL SAVE

4. Add new area

Add Area

Area ID 99	Description amz_gw
IP Address or Range: 0 - 4294967295	
Type Normal	NSSA Translator Role
Default Cost Range: 0 - 16777215	Shortcut Default

Authentication

Auth Type None

ABR Summary Route Filtering

Export List	Import List
Filter List Out	Filter List In

CANCEL ADD

5. Add active interface through which the routes are learned and select the network type as ***point_to_multicast***

Add Interface

Area ID 99	Description
Interface LAN00	Network Type <i>point_to_multicast</i>
<input checked="" type="checkbox"/> Passive Interface	
Advanced Options	
Link Cost 10	DR Priority 1
Default: 10, Range: 1 - 65535	
Retransmit Interval 5	Transmit Delay 1
Default: 5, Range: 1 - 65535	
Hello Interval 10	Dead Interval 40
Default: 10, Range: 1 - 65535	
<input type="checkbox"/> Hello Multiplier 0	<input checked="" type="checkbox"/> Ignore MTU
Range: 1 - 10	
Authentication	
Auth Type None	Auth Key
Digest Key ID	Digest Key
Range: 1 - 255	
CANCEL ADD	

Please Configure OSPF on DC-P with the below mentioned steps:

1. Enable OSPF

The screenshot shows a network configuration interface with a top navigation bar containing tabs for ROUTES, FILTER LISTS, ROUTE MAPS, OSPF, BGP, NAT, MULTICAST, and PORT FORWARDING. The OSPF tab is selected. Below the navigation bar, there is a status bar with icons for STATUS and ROUTE REDISTRIBUTION. A main configuration area displays the Router ID as 0.0.0.0, with an 'Enabled' toggle switch set to 'Enabled'. There are also 'Enable / Disable' and 'ADD AREA' buttons. A table titled 'Areas' is shown, with columns for Area ID, Description, Type, and Auth Type. The table body contains the message 'No Records Found'. At the bottom of the interface, a green notification box indicates 'OSPF Config enabled successfully'.

2. Add the RouterID

OSPF Global Advanced Settings

Options

Router ID 2.2.2.2	Log Adjacency disabled (default)	SPF Delay 0
SPF Hold Time 50	SPF Max Hold Time 5000	Default: 0, Range: 0 - 600000 Default: 5000, Range: 0 - 600000

Default Route Redistribution

<input type="checkbox"/> Redistribute Default	<input type="checkbox"/> Always Redistribute	
Metric 20	Metric Type 2	Route Map
Default: 20, Range: 0 - 16777214	Default: 2, Range: 1 - 2	

Advanced

<input type="checkbox"/> Opaque LSA	<input type="checkbox"/> RFC 1583 Compatibility	
<input type="checkbox"/> Max Metric	Startup Seconds 0	Shutdown Seconds 0
Reference Bandwidth 100000	Write Multiplier 20	ABR Type cisco (default)
Default: 100000, Range: 1 - 4294967	Default: 20, Range: 1 - 100	Default: 0 (Disable), Range: 5 - 86400 Default: 0 (Disable), Range: 5 - 100

Distance Information

CANCEL SAVE

3. Add the redistribution for all kernel routes

Route Redistribution

<input type="checkbox"/> Connected	Metric Range: 0 - 16777214	Metric Type Default: 2, Range: 1 - 2	Route Map	Distribute List
<input checked="" type="checkbox"/> Kernel	Metric 0 Range: 0 - 16777214	Metric Type 2 Default: 2, Range: 1 - 2	Route Map	Distribute List
<input type="checkbox"/> BGP	Metric Range: 0 - 16777214	Metric Type Default: 2, Range: 1 - 2	Route Map	Distribute List
<input type="checkbox"/> Static	Metric Range: 0 - 16777214	Metric Type Default: 2, Range: 1 - 2	Route Map	Distribute List

CANCEL SAVE

4. Add new area

Add Area

Area ID 99	Description amz_gw
IP Address or Range: 0 - 4294967295	
Type Normal	NSSA Translator Role
Default Cost Range: 0 - 16777215	Shortcut Default

Authentication

Auth Type None

ABR Summary Route Filtering

Export List	Import List
Filter List Out	Filter List In

CANCEL ADD

5. Add active interface through which the routes are learned and select the network type as ***point_to_multicast***

Add Interface

Area ID 99	Description
Interface LAN00	Network Type point_to_multicast
<input checked="" type="checkbox"/> Passive Interface	
Advanced Options	
Link Cost 10 Default: 10, Range: 1 - 65535	DR Priority 1 Default: 1, Range: 0 - 255
Retransmit Interval 5 Default: 5, Range: 1 - 65535	Transmit Delay 1 Default: 1, Range: 1 - 65535
Hello Interval 10 Default: 10, Range: 1 - 65535	Dead Interval 40 Default: 40, Range: 1 - 65535
<input type="checkbox"/> Hello Multiplier 0 Range: 1 - 10	<input checked="" type="checkbox"/> Ignore MTU
Authentication	
Auth Type None	Auth Key
Digest Key ID	Digest Key

CANCEL **ADD**

Verify the OSPF config

As shown in the overview, both the branches connects to DC-A, so all the DC Servers should go through DC-A to reach any branch or branch lans.

- OSPF route in AMZ_GATEWAY [External routes are learned from DC-A]

The screenshot shows the Edge Controller interface for the AMZ-LAN-GW device. The navigation bar is at the top with tabs for IPFIX, RSYSLOG, and ROUTER. The ROUTER tab is selected. On the left, a sidebar menu under the 'Protocol' dropdown (set to OSPF) lists OSPF State and Config, OSPF Neighbors, OSPF Neighbor Detail, OSPF Database, OSPF Route, OSPF Interface, and OSPF Border Routers. The main right panel displays the output of the command 'show ip ospf route'. It shows three tables: OSPF network routing table, OSPF router routing table, and OSPF external routing table. The OSPF network routing table contains routes for 10.99.0.1/32, 10.99.0.100/32, 10.99.0.200/32, and 172.67.0.0/24. The OSPF router routing table contains routes for 1.1.1.1 and 2.2.2.2. The OSPF external routing table contains routes for E2 172.8.0.0/16 and E2 172.9.0.0/16.

```

=====
OSPF network routing table
N 10.99.0.1/32      [0] area: 0.0.0.99
                                directly attached to ETH04
N 10.99.0.100/32     [10] area: 0.0.0.99
                                via 10.99.0.100, ETH04
N 10.99.0.200/32     [10] area: 0.0.0.99
                                via 10.99.0.200, ETH04
N 172.67.0.0/24      [10] area: 0.0.0.99
                                directly attached to ETH01

=====
OSPF router routing table
R 1.1.1.1            [10] area: 0.0.0.99, ASBR
                                via 10.99.0.100, ETH04
R 2.2.2.2            [10] area: 0.0.0.99, ASBR
                                via 10.99.0.200, ETH04

=====
OSPF external routing table
N E2 172.8.0.0/16    [10/0] tag: 0
                                via 10.99.0.100, ETH04
N E2 172.9.0.0/16    [10/0] tag: 0
                                via 10.99.0.100, ETH04

```

- OSPF route in DC-A [External routes will be empty, as DC-A is active right now]

The screenshot shows the Edge Controller interface for the DC-A device. The navigation bar is at the top with tabs for IPFIX, RSYSLOG, and ROUTER. The ROUTER tab is selected. On the left, a sidebar menu under the 'Protocol' dropdown (set to OSPF) lists OSPF State and Config, OSPF Neighbors, OSPF Neighbor Detail, OSPF Database, OSPF Route, OSPF Interface, and OSPF Border Routers. The main right panel displays the output of the command 'show ip ospf route'. It shows three tables: OSPF network routing table, OSPF router routing table, and OSPF external routing table. The OSPF network routing table contains routes for 10.99.0.1/32, 10.99.0.100/32, 10.99.0.200/32, and 172.67.0.0/24. The OSPF router routing table contains a route for 2.2.2.2. The OSPF external routing table is empty.

```

=====
OSPF network routing table
N 10.99.0.1/32      [10] area: 0.0.0.99
                                via 10.99.0.1, ETH01
N 10.99.0.100/32     [0] area: 0.0.0.99
                                directly attached to ETH01
N 10.99.0.200/32     [10] area: 0.0.0.99
                                via 10.99.0.200, ETH01
N 172.67.0.0/24      [20] area: 0.0.0.99
                                via 10.99.0.1, ETH01

=====
OSPF router routing table
R 2.2.2.2            [10] area: 0.0.0.99, ASBR
                                via 10.99.0.200, ETH01

=====
OSPF external routing table

```

- OSPF route in DC-B [External routes are learned from DC-A]

The screenshot shows the Edge Controller interface for the DC-B device. The navigation bar is at the top with tabs for IPFIX, RSYSLOG, and ROUTER. The ROUTER tab is selected. On the left, a sidebar menu under the 'Protocol' dropdown (set to OSPF) lists OSPF State and Config, OSPF Neighbors, OSPF Neighbor Detail, OSPF Database, OSPF Route, OSPF Interface, and OSPF Border Routers. The main right panel displays the output of the command 'show ip ospf route'. It shows three tables: OSPF network routing table, OSPF router routing table, and OSPF external routing table. The OSPF network routing table contains routes for 10.99.0.1/32, 10.99.0.100/32, 10.99.0.200/32, and 172.67.0.0/24. The OSPF router routing table contains a route for 1.1.1.1. The OSPF external routing table contains routes for E2 172.8.0.0/16 and E2 172.9.0.0/16.

```

=====
OSPF network routing table
N 10.99.0.1/32      [10] area: 0.0.0.99
                                via 10.99.0.1, ETH01
N 10.99.0.100/32     [10] area: 0.0.0.99
                                via 10.99.0.100, ETH01
N 10.99.0.200/32     [0] area: 0.0.0.99
                                directly attached to ETH01
N 172.67.0.0/24      [20] area: 0.0.0.99
                                via 10.99.0.1, ETH01

=====
OSPF router routing table
R 1.1.1.1            [10] area: 0.0.0.99, ASBR
                                via 10.99.0.100, ETH01

=====
OSPF external routing table
N E2 172.8.0.0/16    [10/0] tag: 0
                                via 10.99.0.100, ETH01
N E2 172.9.0.0/16    [10/0] tag: 0
                                via 10.99.0.100, ETH01

```

Make sure the routes are fine by running IOs

- Check the routes in all the CPEs [AMZ_GATEWAY, DC-A, DC-B, BR-1 & BR-2]

<pre>root@BR-1:/home/router# ip r default nexthop via 10.131.0.99 realm 100001 dev ETH02 weight 1 nexthop via 10.141.0.99 realm 100002 dev ETH03 weight 1 default via 192.168.122.1 dev ETH00 metric 100 10.131.0.8/24 dev IPSEC00 proto kernel scope link src 10.131.0.8 10.131.0.8/24 dev ETH02 proto kernel scope link src 10.131.0.8 10.141.0.8/24 dev IPSEC01 proto kernel scope link src 10.141.0.8 10.141.0.8/24 dev ETH03 proto kernel scope link src 10.141.0.8 172.8.0.8/24 dev ETH04 proto kernel scope link src 172.8.0.8 172.8.0.8/24 nexthop via 10.130.67.1 realm 100003 dev IPSEC00 weight 1 nexthop via 10.140.67.1 realm 100004 dev IPSEC01 weight 1 192.168.122.0/24 dev ETH00 proto kernel scope link src 192.168.122.59 192.168.128.0/28 dev MGMTVPN proto kernel scope link src 192.168.129.8 root@BR-1:/home/router#</pre>	<pre>root@DC-A:/home/router# ip r default nexthop via 10.131.0.99 realm 100001 dev ETH02 weight 1 nexthop via 10.141.0.99 realm 100002 dev ETH03 weight 1 default via 192.168.122.1 dev ETH00 metric 100 10.131.0.8/24 dev IPSEC00 proto kernel scope link src 10.131.0.8 10.131.0.8/24 dev IPSEC01 proto kernel scope link src 10.131.0.8 10.141.0.8/24 dev IPSEC01 proto kernel scope link src 10.140.67.1 10.140.6.8/16 dev ETH03 proto kernel scope link src 10.140.67.1 10.141.0.8/24 dev ETH03 proto kernel scope link src 10.141.0.8 172.8.0.8/16 nexthop via 10.130.67.1 realm 100003 dev IPSEC00 weight 1 nexthop via 10.140.67.1 realm 100004 dev IPSEC01 weight 1 172.9.0.8/16 nexthop via 10.130.9.1 realm 100005 dev IPSEC00 weight 1 nexthop via 10.140.9.1 realm 100006 dev IPSEC01 weight 1 172.168.122.0/24 dev ETH00 proto ospf metric 28 192.168.122.0/24 dev ETH00 proto kernel scope link src 192.168.122.14 192.168.128.0/28 dev MGMTVPN proto kernel scope link src 192.168.129.6 root@DC-A:/home/router#</pre>
<pre>root@BR-2:/home/router# ip r default nexthop via 10.132.0.99 realm 100001 dev ETH02 weight 1 nexthop via 10.142.0.99 realm 100002 dev ETH03 weight 1 default via 192.168.122.1 dev ETH00 metric 100 10.130.0.8/16 dev IPSEC00 proto kernel scope link src 10.130.8.9.1 10.132.0.8/24 dev ETH02 proto kernel scope link src 10.132.0.9.1 10.140.0.8/16 dev IPSEC01 proto kernel scope link src 10.140.8.9.1 10.142.0.8/24 dev ETH03 proto kernel scope link src 10.142.0.9.1 10.142.0.8/24 dev ETH04 proto kernel scope link src 172.9.2.1 172.9.2.0/24 dev ETH04 proto kernel scope link src 172.9.2.1 172.9.2.0/24 nexthop via 10.130.67.1 realm 100003 dev IPSEC00 weight 1 nexthop via 10.140.67.1 realm 100004 dev IPSEC01 weight 1 192.168.122.0/24 dev ETH00 proto kernel scope link src 192.168.122.81 192.168.128.0/28 dev MGMTVPN proto kernel scope link src 192.168.129.9 root@BR-2:/home/router# </pre>	<pre>root@DC-B:/home/router# ip r default nexthop via 10.132.0.99 realm 100001 dev ETH02 weight 1 nexthop via 10.142.0.99 realm 100002 dev ETH03 weight 1 default via 192.168.122.1 dev ETH00 metric 100 10.130.0.8/24 dev IPSEC00 proto kernel scope link src 10.130.8.9.1 10.132.0.8/24 dev ETH02 proto kernel scope link src 10.132.0.9.1 10.140.0.8/16 dev IPSEC01 proto kernel scope link src 10.140.8.9.1 10.142.0.8/24 dev ETH03 proto kernel scope link src 10.142.0.9.1 10.142.0.8/24 dev ETH04 proto kernel scope link src 172.9.2.1 172.9.2.0/24 dev ETH04 proto kernel scope link src 172.9.2.1 172.9.2.0/24 nexthop via 10.130.67.1 realm 100003 dev IPSEC00 weight 1 nexthop via 10.140.67.1 realm 100004 dev IPSEC01 weight 1 192.168.122.0/24 dev ETH00 proto ospf metric 28 192.168.122.0/24 dev ETH00 proto kernel scope link src 192.168.122.61 192.168.128.0/28 dev MGMTVPN proto kernel scope link src 192.168.129.7 root@DC-B:/home/router# </pre>
<pre>root@AMZ-LAN-GW:~# ip r default via 10.141.0.99 dev ETH02 default via 192.168.122.1 dev ETH00 metric 100 10.99.0.8/24 dev ETH04 proto kernel scope link src 10.99.0.8 10.99.0.1 dev ETH04 proto ospf metric 20 10.141.0.8/24 dev ETH02 proto kernel scope link src 10.141.0.5 10.142.0.8/24 dev ETH03 proto kernel scope link src 10.142.0.5 172.8.0.8/16 via 10.99.0.100 dev ETH04 proto ospf metric 20 172.9.0.8/16 via 10.99.0.100 dev ETH04 proto ospf metric 20 172.67.0.8/24 dev ETH01 proto kernel scope link src 172.67.0.1 192.168.122.0/24 dev ETH00 proto kernel scope link src 192.168.122.111 192.168.128.0/28 dev MGMTVPN proto kernel scope link src 192.168.129.5 root@AMZ-LAN-GW:~# </pre>	

- Login to each LAN-PCs and make sure that all got the IPs leased from CPEs

BR-1 LAN PCs:

<pre>router@BR1_LAN1:~\$ ip -br a lo UNKNOWN 127.0.0.1/8 ::1/128 eth0 UP 192.168.122.43/24 fe80::5054:ff:fe82:9ca3/64 eth1 UP 172.8.1.100/24 fe80::acff:fe01:801/64 docker0 DOWN 172.17.8.0/16 router@BR1_LAN1:~\$ ip r default via 172.8.1.1 dev eth1 proto dhcp src 172.8.1.100 metric 100 172.8.1.0/24 dev eth1 proto kernel scope link src 172.8.1.100 172.8.1.1 dev eth1 proto dhcp scope link src 172.8.1.100 metric 100 172.8.1.0/24 dev docker0 proto kernel scope link src 172.17.8.1 linkdown 192.168.122.0/24 dev eth0 proto kernel scope link src 192.168.122.43 router@BR1_LAN1:~\$</pre>	<pre>router@BR1_LAN2:~\$ ip -br a lo UNKNOWN 127.0.0.1/8 ::1/128 eth0 UP 192.168.122.123/24 fe80::5054:ff:fe92:636/64 eth1 UP 172.8.1.101/24 fe80::acff:fe02:801/64 docker0 DOWN 172.17.8.0/16 router@BR1_LAN2:~\$ ip r default via 172.8.1.1 dev eth1 proto dhcp src 172.8.1.101 metric 100 172.8.1.0/24 dev eth1 proto kernel scope link src 172.8.1.101 172.8.1.1 dev eth1 proto dhcp scope link src 172.8.1.101 metric 100 172.17.0.0/16 dev docker0 proto kernel scope link src 172.17.8.1 linkdown 192.168.122.0/24 dev eth0 proto kernel scope link src 192.168.122.123 router@BR1_LAN2:~\$</pre>
<pre>router@BR1_LAN3:~\$ ip -br a lo UNKNOWN 127.0.0.1/8 ::1/128 eth0 UP 192.168.122.36/24 fe80::5054:ff:fe43:fua2/64 eth1 UP 172.8.2.100/24 fe80::acff:fe03:801/64 docker0 DOWN 172.17.8.0/16 router@BR1_LAN3:~\$ ip r default via 172.8.2.1 dev eth1 proto dhcp src 172.8.2.100 metric 100 172.8.2.0/24 dev eth1 proto kernel scope link src 172.8.2.100 172.8.2.1 dev eth1 proto dhcp scope link src 172.8.2.100 metric 100 172.17.0.0/16 dev docker0 proto kernel scope link src 172.17.8.1 linkdown 192.168.122.0/24 dev eth0 proto kernel scope link src 192.168.122.30 router@BR1_LAN3:~\$ </pre>	<pre>router@BR1_LAN4:~\$ ip -br a lo UNKNOWN 127.0.0.1/8 ::1/128 eth0 UP 192.168.122.33/24 fe80::5054:ff:fe5a:41a7/64 eth1 UP 172.8.2.101/24 fe80::acff:fe04:801/64 docker0 DOWN 172.17.8.0/16 router@BR1_LAN4:~\$ ip r default via 172.8.2.1 dev eth1 proto dhcp src 172.8.2.101 metric 100 172.8.2.0/24 dev eth1 proto kernel scope link src 172.8.2.101 172.8.2.1 dev eth1 proto dhcp scope link src 172.8.2.101 metric 100 172.17.0.0/16 dev docker0 proto kernel scope link src 172.17.8.1 linkdown 192.168.122.0/24 dev eth0 proto kernel scope link src 192.168.122.33 router@BR1_LAN4:~\$ </pre>

BR-2 LAN PCs:

<pre>router@BR2_LAN1:\$ ip -br a lo UNKNOWN 127.0.0.1/8 ::1/128 eth0 UP 192.168.122.51/24 fe80::5054:ff:fe94:d3d5/64 eth1 UP 172.9.1.100/24 fe80::acff:fe01:981/64 docker0 DOWN 172.17.0.1/16 router@BR2_LAN1:\$ ip r default via 172.9.1.1 dev eth1 default via 172.9.2.1 dev eth1 proto dhcp src 172.9.2.100 metric 100 172.9.2.0/24 dev eth1 proto kernel scope link src 172.9.2.100 172.9.2.1 dev eth1 proto dhcp scope link src 172.9.2.100 metric 100 172.17.0.0/16 dev docker0 proto kernel scope link src 172.17.0.1 linkdown 192.168.122.0/24 dev eth0 proto kernel scope link src 192.168.122.121 router@BR2_LAN1:\$ </pre>	<pre>router@BR2_LAN4:\$ ip -br a lo UNKNOWN 127.0.0.1/8 ::1/128 eth0 UP 192.168.122.20/24 fe80::5054:ff:fe70:64 eth1 UP 172.9.2.101/24 fe80::acff:fe01:901/64 docker0 DOWN 172.17.0.1/16 router@BR2_LAN4:\$ ip r default via 172.9.2.1 dev eth1 proto dhcp src 172.9.2.101 metric 100 172.9.2.0/24 dev eth1 proto kernel scope link src 172.9.2.101 172.9.2.1 dev eth1 proto dhcp scope link src 172.9.2.101 metric 100 172.17.0.0/16 dev docker0 proto kernel scope link src 172.17.0.1 linkdown 192.168.122.0/24 dev eth0 proto kernel scope link src 192.168.122.29 router@BR2_LAN4:\$</pre>
---	--

DC LAN Servers:

<pre>router@DC_LAN1:\$ ip -br a lo UNKNOWN 127.0.0.1/8 ::1/128 eth0 UP 192.168.122.26/24 fe80::5054:ff:fea3:3274/64 eth1 UP 172.67.0.100/24 fe80::acff:fe01:501/64 docker0 DOWN 172.17.0.1/16 router@DC_LAN1:\$ ip r default via 172.67.0.1 dev eth1 default via 172.67.0.1 dev eth1 proto dhcp src 172.67.0.100 metric 100 172.17.0.0/16 dev docker0 proto kernel scope link src 172.17.0.1 linkdown 172.67.0.0/24 dev eth1 proto kernel scope link src 172.67.0.100 172.67.0.1 dev eth1 proto dhcp scope link src 172.67.0.100 metric 100 192.168.122.0/24 dev eth0 proto kernel scope link src 192.168.122.26 router@DC_LAN1:\$ </pre>	<pre>router@DC_LAN2:\$ ip -br a lo UNKNOWN 127.0.0.1/8 ::1/128 eth0 UP 192.168.122.119/24 fe80::5054:ff:feb6:b5c7/64 eth1 UP 172.67.0.103/24 fe80::acff:fe04:501/64 docker0 DOWN 172.17.0.1/16 router@DC_LAN2:\$ ip r default via 172.67.0.1 dev eth1 default via 172.67.0.1 dev eth1 proto dhcp src 172.67.0.103 metric 100 172.17.0.0/16 dev docker0 proto kernel scope link src 172.17.0.1 linkdown 172.67.0.0/24 dev eth1 proto kernel scope link src 172.67.0.103 172.67.0.1 dev eth1 proto dhcp scope link src 172.67.0.103 metric 100 192.168.122.0/24 dev eth0 proto kernel scope link src 192.168.122.119 router@DC_LAN2:\$ </pre>
<pre>router@DC_LAN3:\$ ip -br a lo UNKNOWN 127.0.0.1/8 ::1/128 eth0 UP 192.168.122.48/24 fe80::5054:ff:fea5:c204/64 eth1 UP 172.67.0.102/24 fe80::acff:fe03:501/64 docker0 DOWN 172.17.0.1/16 router@DC_LAN3:\$ ip r default via 172.67.0.1 dev eth1 default via 172.67.0.1 dev eth1 proto dhcp src 172.67.0.102 metric 100 172.17.0.0/16 dev docker0 proto kernel scope link src 172.17.0.1 linkdown 172.67.0.0/24 dev eth1 proto kernel scope link src 172.67.0.102 172.67.0.1 dev eth1 proto dhcp scope link src 172.67.0.102 metric 100 192.168.122.0/24 dev eth0 proto kernel scope link src 192.168.122.48 router@DC_LAN3:\$ </pre>	<pre>router@DC_LAN4:\$ ip -br a lo UNKNOWN 127.0.0.1/8 ::1/128 eth0 UP 192.168.122.119/24 fe80::5054:ff:feb6:b5c7/64 eth1 UP 172.67.0.103/24 fe80::acff:fe04:501/64 docker0 DOWN 172.17.0.1/16 router@DC_LAN4:\$ ip r default via 172.67.0.1 dev eth1 default via 172.67.0.1 dev eth1 proto dhcp src 172.67.0.103 metric 100 172.17.0.0/16 dev docker0 proto kernel scope link src 172.17.0.1 linkdown 172.67.0.0/24 dev eth1 proto kernel scope link src 172.67.0.103 172.67.0.1 dev eth1 proto dhcp scope link src 172.67.0.103 metric 100 192.168.122.0/24 dev eth0 proto kernel scope link src 192.168.122.119 router@DC_LAN4:\$ </pre>

- Start the IOs from all the branch lan pcs and monitor the same using tcpdump in dc lan servers

BR-1 LAN running IOs:

BR-2 LAN running IOs:

```
router@BR2_LAN1:~$ ping 172.67.8.183
PING 172.67.8.183 (172.67.8.183) 56(84) bytes of data.
64 bytes from 172.67.8.183: icmp_seq=1 ttl=61 time=5.36 ms
64 bytes from 172.67.8.183: icmp_seq=2 ttl=61 time=11.8 ms
64 bytes from 172.67.8.183: icmp_seq=3 ttl=61 time=10.8 ms
64 bytes from 172.67.8.183: icmp_seq=4 ttl=61 time=4.46 ms
64 bytes from 172.67.8.183: icmp_seq=5 ttl=61 time=4.46 ms
64 bytes from 172.67.8.183: icmp_seq=6 ttl=61 time=4.54 ms
64 bytes from 172.67.8.183: icmp_seq=7 ttl=61 time=12.8 ms

router@BR2_LAN2:~$ ping 172.67.8.182
PING 172.67.8.182 (172.67.8.182) 56(84) bytes of data.
64 bytes from 172.67.8.182: icmp_seq=1 ttl=61 time=0.59 ms
64 bytes from 172.67.8.182: icmp_seq=2 ttl=61 time=0.88 ms
64 bytes from 172.67.8.182: icmp_seq=3 ttl=61 time=0.88 ms
64 bytes from 172.67.8.182: icmp_seq=4 ttl=61 time=0.19 ms
64 bytes from 172.67.8.182: icmp_seq=5 ttl=61 time=0.46 ms
64 bytes from 172.67.8.182: icmp_seq=6 ttl=61 time=3.99 ms

router@BR2_LAN3:~$ ping 172.67.8.101
PING 172.67.8.101 (172.67.8.101) 56(84) bytes of data.
64 bytes from 172.67.8.101: icmp_seq=1 ttl=61 time=5.24 ms
64 bytes from 172.67.8.101: icmp_seq=2 ttl=61 time=4.48 ms
64 bytes from 172.67.8.101: icmp_seq=3 ttl=61 time=4.52 ms
64 bytes from 172.67.8.101: icmp_seq=4 ttl=61 time=4.34 ms
64 bytes from 172.67.8.101: icmp_seq=5 ttl=61 time=3.96 ms

router@BR2_LAN4:~$ ping 172.67.8.100
PING 172.67.8.100 (172.67.8.100) 56(84) bytes of data.
64 bytes from 172.67.8.100: icmp_seq=1 ttl=61 time=0.14 ms
64 bytes from 172.67.8.100: icmp_seq=2 ttl=61 time=0.25 ms
64 bytes from 172.67.8.100: icmp_seq=3 ttl=61 time=0.12 ms
64 bytes from 172.67.8.100: icmp_seq=4 ttl=61 time=0.13 ms
64 bytes from 172.67.8.100: icmp_seq=5 ttl=61 time=0.23 ms
```

DC LAN running tcpdumps:

```
root@DC_LAN1:/home/router# tcpdump -i eth1 -n icmp
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth1, link-type EN10MB (Ethernet), capture size 262144 bytes
18:30:59.509888 IP 172.9.2.101 > 172.67.0.100: ICMP echo request, id 27721, seq 136, length 64
18:30:59.510089 IP 172.9.2.101 > 172.67.0.100: ICMP echo reply, id 27721, seq 136, length 64
18:30:59.522075 IP 172.9.2.101 > 172.67.0.100: ICMP echo request, id 28292, seq 85, length 64
18:31:00.222969 IP 172.9.2.100 > 172.67.0.100: ICMP echo reply, id 28292, seq 85, length 64
18:31:00.512969 IP 172.9.2.100 > 172.67.0.100: ICMP echo request, id 27721, seq 137, length 64
18:31:00.512983 IP 172.9.2.100 > 172.67.0.100: ICMP echo reply, id 27721, seq 137, length 64
18:31:00.512997 IP 172.9.2.100 > 172.67.0.100: ICMP echo request, id 28292, seq 86, length 64
18:31:01.231390 IP 172.9.2.100 > 172.67.0.100: ICMP echo reply, id 28292, seq 86, length 64
18:31:01.512897 IP 172.9.2.101 > 172.67.0.100: ICMP echo request, id 27721, seq 138, length 64
18:31:01.512909 IP 172.9.2.101 > 172.67.0.100: ICMP echo reply, id 27721, seq 138, length 64
18:31:02.233099 IP 172.9.2.100 > 172.67.0.100: ICMP echo request, id 27721, seq 139, length 64
18:31:02.233166 IP 172.9.2.100 > 172.67.0.100: ICMP echo reply, id 27721, seq 139, length 64
18:31:02.512932 IP 172.9.2.100 > 172.67.0.100: ICMP echo request, id 28292, seq 87, length 64
18:31:02.512946 IP 172.9.2.100 > 172.67.0.100: ICMP echo reply, id 28292, seq 87, length 64
18:31:03.235081 IP 172.9.2.100 > 172.67.0.100: ICMP echo request, id 28292, seq 88, length 64
18:31:03.235104 IP 172.9.2.100 > 172.67.0.100: ICMP echo reply, id 28292, seq 88, length 64
```

```
root@DC_LAN2:/home/router# tcpdump -i eth1 -n icmp
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth1, link-type EN10MB (Ethernet), capture size 262144 bytes
18:31:00.004497 IP 172.8.1.101 > 172.67.0.101: ICMP echo request, id 28314, seq 84, length 64
18:31:00.004501 IP 172.8.1.101 > 172.67.0.101: ICMP echo reply, id 28314, seq 84, length 64
18:31:00.00739427 IP 172.8.1.100 > 172.67.0.101: ICMP echo request, id 27566, seq 139, length 64
18:31:00.739585 IP 172.8.1.100 > 172.67.0.101: ICMP echo reply, id 27566, seq 138, length 64
18:31:00.739585 IP 172.8.1.101 > 172.67.0.101: ICMP echo request, id 28314, seq 85, length 64
18:31:01.004608 IP 172.8.1.101 > 172.67.0.101: ICMP echo reply, id 28314, seq 85, length 64
18:31:01.004608 IP 172.8.1.101 > 172.67.0.101: ICMP echo request, id 27566, seq 139, length 64
18:31:01.740127 IP 172.8.1.101 > 172.67.0.101: ICMP echo reply, id 27566, seq 139, length 64
18:31:01.740204 IP 172.8.1.101 > 172.67.0.101: ICMP echo request, id 28314, seq 86, length 64
18:31:02.0048615 IP 172.8.1.101 > 172.67.0.101: ICMP echo request, id 28314, seq 86, length 64
18:31:02.0048693 IP 172.8.1.101 > 172.67.0.101: ICMP echo reply, id 28314, seq 86, length 64
18:31:02.748341 IP 172.8.1.101 > 172.67.0.101: ICMP echo request, id 27566, seq 140, length 64
18:31:02.748422 IP 172.8.1.101 > 172.67.0.101: ICMP echo reply, id 27566, seq 140, length 64
18:31:02.8049546 IP 172.8.1.101 > 172.67.0.101: ICMP echo request, id 28314, seq 87, length 64
18:31:03.0049516 IP 172.8.1.101 > 172.67.0.101: ICMP echo reply, id 28314, seq 87, length 64
```

```
root@DC_LAN3:/home/router# tcpdump -i eth1 -n icmp
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth1, link-type EN10MB (Ethernet), capture size 262144 bytes
18:31:00.699811 IP 172.8.2.100 > 172.67.0.102: ICMP echo request, id 27205, seq 84, length 64
18:31:00.699889 IP 172.67.0.102 > 172.8.2.100: ICMP echo reply, id 27205, seq 84, length 64
18:31:01.0040338 IP 172.9.1.101 > 172.67.0.102: ICMP echo request, id 27394, seq 139, length 64
18:31:01.0040424 IP 172.9.1.101 > 172.67.0.102: ICMP echo request, id 27394, seq 139, length 64
18:31:01.0040429 IP 172.9.1.101 > 172.67.0.102: ICMP echo request, id 27205, seq 85, length 64
18:31:01.0040431 IP 172.9.1.101 > 172.67.0.102: ICMP echo request, id 27205, seq 85, length 64
18:31:01.731713 IP 172.67.0.102 > 172.9.1.101: ICMP echo reply, id 27205, seq 85, length 64
18:31:02.050005 IP 172.9.1.101 > 172.67.0.102: ICMP echo request, id 27394, seq 140, length 64
18:31:02.050081 IP 172.67.0.102 > 172.9.1.101: ICMP echo reply, id 27394, seq 140, length 64
18:31:02.7108811 IP 172.8.2.100 > 172.67.0.102: ICMP echo request, id 27205, seq 86, length 64
18:31:02.7108899 IP 172.67.0.102 > 172.8.2.100: ICMP echo reply, id 27205, seq 86, length 64
18:31:03.0047176 IP 172.9.1.101 > 172.67.0.102: ICMP echo request, id 27394, seq 141, length 64
18:31:03.0047249 IP 172.67.0.102 > 172.9.1.101: ICMP echo reply, id 27394, seq 141, length 64
```

```
root@DC_LAN4:/home/router# tcpdump -i eth1 -n icmp
tcpdump: verbose output suppressed, use -v or -vv for full protocol decode
listening on eth1, link-type EN10MB (Ethernet), capture size 262144 bytes
18:31:01.374007 IP 172.8.2.101 > 172.67.0.103: ICMP echo request, id 28770, seq 84, length 64
18:31:01.374138 IP 172.67.0.103 > 172.8.2.101: ICMP echo reply, id 28770, seq 84, length 64
18:31:01.425741 IP 172.9.1.100 > 172.67.0.103: ICMP echo request, id 26891, seq 140, length 64
18:31:01.425810 IP 172.67.0.103 > 172.9.1.100: ICMP echo reply, id 26891, seq 140, length 64
18:31:01.425837 IP 172.9.1.100 > 172.67.0.103: ICMP echo request, id 28770, seq 85, length 64
18:31:01.425840 IP 172.67.0.103 > 172.9.1.100: ICMP echo reply, id 28770, seq 85, length 64
18:31:02.427305 IP 172.9.1.100 > 172.67.0.103: ICMP echo request, id 26891, seq 141, length 64
18:31:02.427305 IP 172.67.0.103 > 172.9.1.100: ICMP echo reply, id 26891, seq 141, length 64
18:31:02.427397 IP 172.67.0.103 > 172.9.1.100: ICMP echo request, id 28770, seq 86, length 64
18:31:03.385888 IP 172.8.2.101 > 172.67.0.103: ICMP echo request, id 28770, seq 86, length 64
18:31:03.385937 IP 172.67.0.103 > 172.8.2.101: ICMP echo reply, id 28770, seq 86, length 64
18:31:03.436982 IP 172.9.1.100 > 172.67.0.103: ICMP echo request, id 26891, seq 142, length 64
18:31:03.437069 IP 172.67.0.103 > 172.9.1.100: ICMP echo reply, id 26891, seq 142, length 64
```