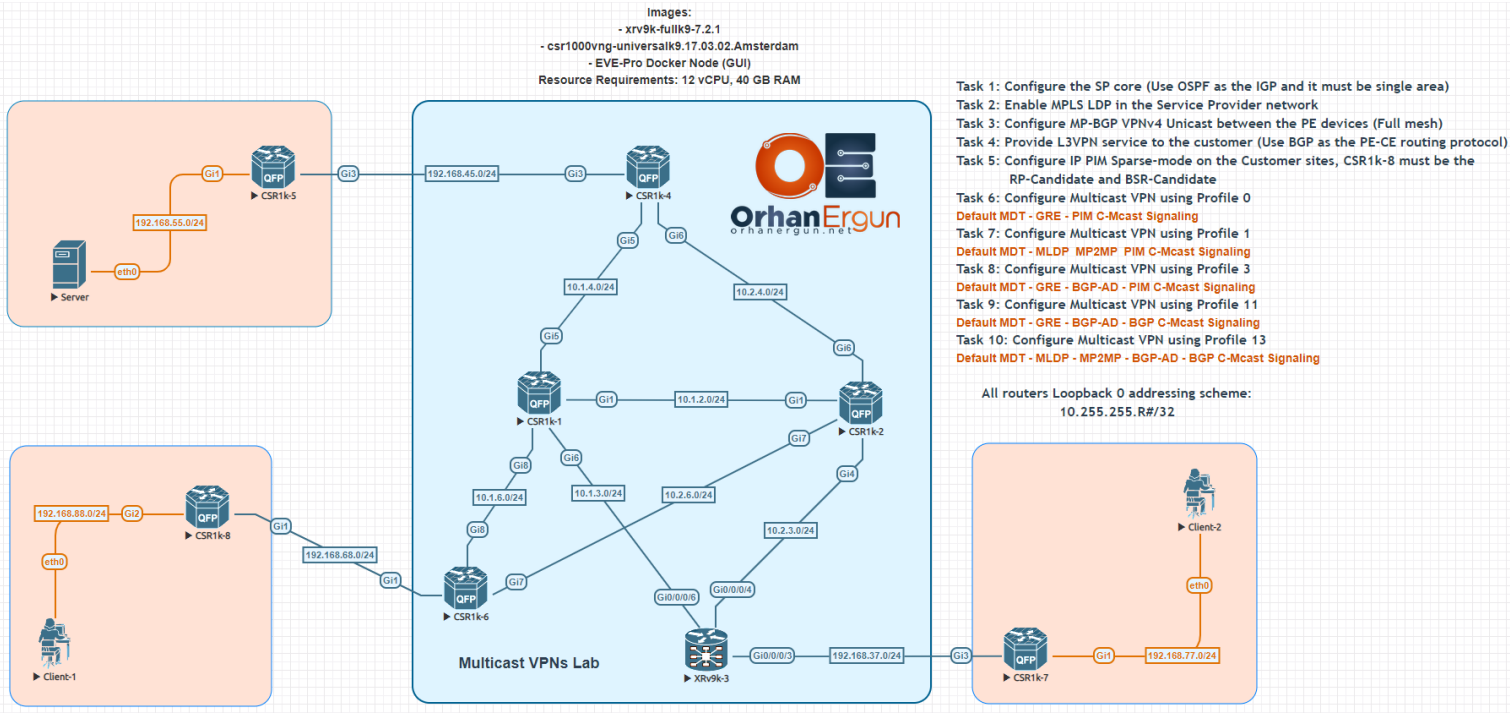
mVPN Lab

Multicast Virtual Private Networks

Topology:



MP-BGP AF/SAF: VPNv4 Unicast, IPv4 MDT, IPv4 MVPN

L3VPN PE-CE Routing Protocol:

- BGP

Multicast routing protocol:

- PIM (Sparse-mode)

Task 01 - 04:

- Configure the SP core (Use OSPF as the IGP and it must be single area)
- Enable MPLS LDP in the Service Provider network
- Configure MP-BGP VPNv4 Unicast between the PE devices (Full mesh)
- Provide L3VPN service to the customer (Use BGP as the PE-CE routing protocol)
- Configure IP PIM Sparse-mode on the Customer sites, CSR1k-8 must be the

Solution:

MVPN stands for Multicast VPN, when a service provider needs to also deal with customer multicast traffic (in the case of L3VPN, which means, service provider, provides L3VPN service to the customer), the service provider can use different methods to provide this kind of service.

It is obvious that if a service provider, provides L2VPNs services like VPLS or EVPN, there is no need to do any additional steps on the service provider side to carry multicast traffic in addition to the unicast traffic, because in those services any BUM traffic will be forwarded to the other customer sites.

So when we talk about MVPN, we are mentioning Multicast service in addition to the L3VPN service.

There are different Multicast VPN implementation profiles that are actually a combination of different protocols or encapsulations and signaling to deal with the customer traffic (Take it and tunnel it to the other sites or the sites that need that traffic).

In all of the Multicast VPNs profiles, L3VPN service should be already configured. Task 01 – Task 04 is just focusing on configuring the service provider core and getting it ready to the L3VPN service and also providing the L3VPN service for the customer.

We are not going to provide explanation about these 4 initial tasks, if you need more information or you don't feel comfortable configuring L3VPN service, please refer to the MPLS Lab 01 and MPLS Lab 02, then you can come back to this lab and complete it.

In these 4 initial tasks, we will configure single area OSPF in the SP core, enable MPLS LDP, configure MP-BGP VPNv4 unicast neighborship in a full mesh manner between PE devices (There is no Route-Reflector in the topology) and finaly provide L3VPN service for the customer (PE-CE Routing protocol: BGP).

Let's get started:

CSR1K-1:

```
router ospf 1000
 router-id 0.0.0.1
 auto-cost reference-bandwidth 100000
 passive-interface Loopback0
 mpls ldp autoconfig
!
mpls label range 1000 1999
mpls ldp router-id Loopback0
!
interface GigabitEthernet1
 ip ospf network point-to-point
 ip ospf 1000 area 0
!
interface GigabitEthernet5
 ip ospf network point-to-point
 ip ospf 1000 area 0
!
interface GigabitEthernet6
 ip ospf network point-to-point
 ip ospf 1000 area 0
!
interface GigabitEthernet8
 ip ospf network point-to-point
 ip ospf 1000 area 0
!
```

CSR1K-2:

```
router ospf 1000
 router-id 0.0.0.2
 auto-cost reference-bandwidth 100000
 passive-interface Loopback0
 mpls ldp autoconfig
!
mpls label range 2000 2999
mpls ldp router-id Loopback0
```

```
!
interface GigabitEthernet1
 ip ospf network point-to-point
 ip ospf 1000 area 0
!
interface GigabitEthernet4
 ip ospf network point-to-point
 ip ospf 1000 area 0
!
interface GigabitEthernet6
 ip ospf network point-to-point
 ip ospf 1000 area 0
!
interface GigabitEthernet7
 ip ospf network point-to-point
 ip ospf 1000 area 0
!
```

CSR1K-4:

```
router ospf 1000
 router-id 0.0.0.4
 auto-cost reference-bandwidth 100000
 passive-interface Loopback0
 mpls ldp autoconfig
!
interface GigabitEthernet5
 ip ospf network point-to-point
 ip ospf 1000 area 0
!
interface GigabitEthernet6
 ip ospf network point-to-point
 ip ospf 1000 area 0
!
```

CSR1K-6:

```
router ospf 1000
 router-id 0.0.0.6
 auto-cost reference-bandwidth 100000
 passive-interface Loopback0
 mpls ldp autoconfig
!
interface GigabitEthernet7
 ip ospf network point-to-point
 ip ospf 1000 area 0
!
interface GigabitEthernet8
 ip ospf network point-to-point
 ip ospf 1000 area 0
!
```

```
XRv9k-3:
mpls ldp
 router-id 10.255.255.3
!
mpls label range table 0 30000 39999
!
router ospf 1000
 log adjacency changes detail
 router-id 0.0.0.3
 mpls ldp auto-config
 address-family ipv4
 area 0
  interface Loopback0
   passive enable
  !
  interface GigabitEthernet0/0/0/4
   network point-to-point
  !
  interface GigabitEthernet0/0/0/6
```

```
   network point-to-point
  !
 !
!
commit
!
```

Basic IGP and MPLS LDP configuration has been done.

Let's configure BGP on the PE devices and provide L3VPN service to the customer:

```
CSR1K-4:
vrf definition Customer1
 rd 1000:1001
 !
 address-family ipv4
  route-target export 1000:1001
  route-target import 1000:1001
 exit-address-family
!
!
interface GigabitEthernet3
 vrf forwarding Customer1
 ip address 192.168.45.4 255.255.255.0
!
router bgp 1000
 bgp log-neighbor-changes
 no bgp default ipv4-unicast
 neighbor 10.255.255.3 remote-as 1000
 neighbor 10.255.255.3 update-source Loopback0
 neighbor 10.255.255.6 remote-as 1000
 neighbor 10.255.255.6 update-source Loopback0
 !
 address-family ipv4
 exit-address-family
 !
 address-family vpnv4
```

```
  neighbor 10.255.255.3 activate
  neighbor 10.255.255.3 send-community extended
  neighbor 10.255.255.6 activate
  neighbor 10.255.255.6 send-community extended
 exit-address-family
 !
 address-family ipv4 vrf Customer1
  redistribute connected metric 20000
  neighbor 192.168.45.5 remote-as 65001
  neighbor 192.168.45.5 activate
  neighbor 192.168.45.5 as-override
 exit-address-family
!
```

CSR1k-6:

```
vrf definition Customer1
 rd 1000:1001
 !
 address-family ipv4
  route-target export 1000:1001
  route-target import 1000:1001
 exit-address-family
!
!
interface GigabitEthernet1
 vrf forwarding Customer1
 ip address 192.168.68.6 255.255.255.0
!
router bgp 1000
 bgp log-neighbor-changes
 no bgp default ipv4-unicast
 neighbor 10.255.255.3 remote-as 1000
 neighbor 10.255.255.3 update-source Loopback0
 neighbor 10.255.255.4 remote-as 1000
 neighbor 10.255.255.4 update-source Loopback0
```

```
 !
 address-family ipv4
 exit-address-family
 !
 address-family vpnv4
  neighbor 10.255.255.3 activate
  neighbor 10.255.255.3 send-community extended
  neighbor 10.255.255.4 activate
  neighbor 10.255.255.4 send-community extended
 exit-address-family
 !
 address-family ipv4 vrf Customer1
  redistribute connected metric 20000
  neighbor 192.168.68.8 remote-as 65001
  neighbor 192.168.68.8 activate
  neighbor 192.168.68.8 as-override
 exit-address-family
!
```

XRv9k-3:

```
vrf Customer1
 address-family ipv4 unicast
  import route-target
   1000:1001
  !
  export route-target
   1000:1001
  !
 !
!
route-policy eBGP-PASS
  pass
end-policy
!
```

```
router bgp 1000
 bgp log neighbor changes detail
 address-family vpnv4 unicast
 !
 address-family ipv4 mdt
 !
 neighbor 10.255.255.4
  remote-as 1000
  update-source Loopback0
  address-family vpnv4 unicast
  !
  address-family ipv4 mdt
  !
 !
 neighbor 10.255.255.6
  remote-as 1000
  update-source Loopback0
  address-family vpnv4 unicast
  !
!
 vrf Customer1
  rd 1000:1001
  address-family ipv4 unicast
   redistribute connected metric 20000
  !
  neighbor 192.168.37.7
   remote-as 65001
   address-family ipv4 unicast
    route-policy eBGP-PASS in
    route-policy eBGP-PASS out
    as-override
   !
  !
 !
!
```

```
router bgp 1000
 bgp log neighbor changes detail
 address-family vpnv4 unicast
 !
 address-family ipv4 mdt
 !
 neighbor 10.255.255.4
  remote-as 1000
  update-source Loopback0
  address-family vpnv4 unicast
  !
  address-family ipv4 mdt
  !
 !
 neighbor 10.255.255.6
  remote-as 1000
  update-source Loopback0
  address-family vpnv4 unicast
  !
!
 vrf Customer1
  rd 1000:1001
  address-family ipv4 unicast
   redistribute connected metric 20000
  !
  neighbor 192.168.37.7
   remote-as 65001
   address-family ipv4 unicast
    route-policy eBGP-PASS in
    route-policy eBGP-PASS out
    as-override
   !
  !
commit
!
```

PE routers configuration done.

Time to get into the Customer CE devices and do the necessary configuration on them:

CSR1K-5:

```
router bgp 65001
 bgp log-neighbor-changes
 network 10.255.255.5 mask 255.255.255.255
 network 192.168.55.0
 neighbor 192.168.45.4 remote-as 1000
!
```

CSR1K-8:

```
router bgp 65001
 bgp log-neighbor-changes
 network 10.255.255.8 mask 255.255.255.255
 network 192.168.88.0
 neighbor 192.168.68.6 remote-as 1000
!
```

CSR1K-7:

```
router bgp 65001
 bgp log-neighbor-changes
 network 10.255.255.7 mask 255.255.255.255
 network 192.168.77.0
 neighbor 192.168.37.3 remote-as 1000
!
```

Verification:

```
CSR1k-5#show bgp ipv4 uni  | begin Net
     Network          Next Hop            Metric LocPrf Weight Path
 *>   10.255.255.5/32  0.0.0.0                  0         32768 i
 *>   10.255.255.7/32  192.168.45.4                           0 1000 1000 i
 *>   10.255.255.8/32  192.168.45.4                           0 1000 1000 i
 *>   192.168.37.0     192.168.45.4                           0 1000 ?
 r>   192.168.45.0     192.168.45.4         20000             0 1000 ?
 *>   192.168.55.0     0.0.0.0                  0         32768 i
```

```
 *>   192.168.68.0     192.168.45.4                           0 1000 ?
 *>   192.168.77.0     192.168.45.4                           0 1000 1000 i
 *>   192.168.88.0     192.168.45.4                           0 1000 1000


CSR1k-5#traceroute 10.255.255.7 source lo 0
Type escape sequence to abort.
Tracing the route to 10.255.255.7
VRF info: (vrf in name/id, vrf out name/id)
  1 192.168.45.4 [AS 1000] 7 msec 1 msec 1 msec
  2 10.1.4.1 [MPLS: Labels 1005/30013 Exp 0] 14 msec 3 msec 2 msec
  3 10.1.3.3 [MPLS: Label 30013 Exp 0] 8 msec 2 msec 2 msec
  4 192.168.37.7 [AS 1000] 3 msec 2 msec 2 msec
CSR1k-5#traceroute 10.255.255.8 source lo 0
Type escape sequence to abort.
Tracing the route to 10.255.255.8
VRF info: (vrf in name/id, vrf out name/id)
  1 192.168.45.4 [AS 1000] 6 msec 4 msec 1 msec
  2 10.2.4.2 [MPLS: Labels 2007/6011 Exp 0] 4 msec 2 msec 1 msec
  3 192.168.68.6 [AS 1000] [MPLS: Label 6011 Exp 0] 2 msec 2 msec 2 msec
  4 192.168.68.8 [AS 1000] 2 msec 1 msec 2 msec
```

Everything works fine as we expected!

Task 05:

- Configure IP PIM Sparse-mode on the Customer sites, CSR1k-8 must be the RP-Candidate and BSR-Candidate

Solution:

In all MVPN profiles, there is need to configure PIM on the customer side and also on the link that connects PE device to the customer device (PE-CE physical interfaces).

In the entire lab, we will use PIM Sparse mode:

```
CSR1K-5:
ip multicast-routing distributed
interface GigabitEthernet1
 ip address 192.168.55.254 255.255.255.0
 ip pim sparse-mode
!
interface GigabitEthernet3
 ip address 192.168.45.5 255.255.255.0
 ip pim sparse-mode
!

CSR1K-8:
ip multicast-routing distributed
interface GigabitEthernet2
 ip address 192.168.88.254 255.255.255.0
 ip pim sparse-mode
!
interface GigabitEthernet1
 ip address 192.168.68.8 255.255.255.0
 ip pim sparse-mode
!

CSR1K-7:
ip multicast-routing distributed
interface GigabitEthernet1
```

```
 ip address 192.168.77.7 255.255.255.0
 ip pim sparse-mode
!
interface GigabitEthernet3
 ip address 192.168.37.7 255.255.255.0
 ip pim sparse-mode
!
```

The task also asked us to configure CSR1K-8 as being the RP-Candidate and BSR-Candidate:

```
CSR1k-8:
interface Loopback0
 ip address 10.255.255.8 255.255.255.255
 ip pim sparse-mode
!
ip pim bsr-candidate Loopback0 0
ip pim rp-candidate Loopback0
!
```

In PIM Sparse mode, it is a must to have an RP, we could configure it manualy on each of the routers, but the task expilicitly mentioned that BSR should be configured.

Task 06:

- Configure Multicast VPN using Profile 0
  Default MDT – GRE – PIM C-Mcast Signaling

Solution:

What is MVPN Profiles? Do we need to remember them?

Actually there is no need to remember the profile configuration templates. You can find them easily on the cisco website. The reason is: there are too many of them! They are just different combination of protocols and their interaction.

IOS-XE mVPN Profiles:

https://www.cisco.com/c/en/us/support/docs/ip/multicast/118985-configure-mcast-00.html

IOS-XR mVPN Profiles:

https://www.cisco.com/c/en/us/support/docs/ip/multicast/200512-Configure-mVPN-Profiles-within-Cisco-IOS.html

You can just follow the config examples in order to do the necessary configuration on your devices.

In Profile 0 we have this description:

- Default MDT – GRE – PIM C-Mcast Signaling

This Profile is using:

- Default MDT (Multicast Distribution Tree) between PE devices
- GRE for encapsulating the customer multicast traffic and send it to other PE devices (in fact this is mGRE)
- PIM C-Mcast Signaling means: using PIM in order to signal the Customer multicast routes (*, G) (S,G) etc… .

How these PE devices can find out about each other? How they can build MDT (Multicast Distribution Tree) when they are remote from each other? (There are two core devices at the middle of the topology, so PE devices are not directly connected to each other)?

The answer is: MP-BGP again! (we will also discuss about the other method soon).

There are two MP-BGP AF/SAF that can help PE devices to build the MDT and Discover each other. Those AF/SAFs are:

- IPv4 MDT
- IPv4 MVPN

The only profile that uses IPv4 MDT is profile 0 (it is some kind of old fashion AF/SAF, nowadays we prefer to use IPv4 mVPN), when there is need to have Auto-Discovery, BGP comes into play to help us!

Let's start the configuration:

```
CSR1k-4:
ip multicast-routing distributed
ip multicast-routing vrf Customer1 distributed
interface GigabitEthernet3
 vrf forwarding Customer1
 ip address 192.168.45.4 255.255.255.0
 ip pim sparse-mode
!
router bgp 1000
 address-family ipv4 mdt
  neighbor 10.255.255.3 activate
  neighbor 10.255.255.3 send-community extended
  neighbor 10.255.255.6 activate
  neighbor 10.255.255.6 send-community extended
 exit-address-family
!
vrf definition Customer1
!
 address-family ipv4
  mdt default 232.1.1.1
!
interface GigabitEthernet5
 ip pim sparse-mode
!
interface GigabitEthernet6
 ip pim sparse-mode
!
```

As you relized: IPv4 MDT address family has been enabled for two neighbors (two PE devices) and IPv4 Multicast routing has been enable both on GRT and the VRF.

Under the vrf definition configuration we have also configured mdt default 232.1.1.1.

This is the Multicast Group address that we specified for this customer traffic. When there is a need to forward the multicast traffic to the other sites, the PE will encapsulate the Customer multicast traffic and forwards it to the core of the network, all the PE devices connected to the customer sites is going to join to this group, so they want to receive those traffic.

Let's configure the rest of the device and in the verification section, you will realize that why we need these information:

```
CSR1k-4:
interface Loopback0
 ip address 10.255.255.4 255.255.255.255
 ip pim sparse-mode
!
```

This is also an important part of the configuration! As we mentioned before the mGRE is going to be used in order to encapsulate the customer multicast traffic. This mGRE should be formed using some interface (Loopback interface, because we have used Loopback interface as the update-source of the MP-BGP neighborship), that is why you need to enable ip pim sparse-mode on this Loopback interface, otherwise it won't work, the traffic is not going to be encapsulated inside GRE.

```
CSR1k-6:
ip multicast-routing distributed
ip multicast-routing vrf Customer1 distributed
vrf definition Customer1
 !
 address-family ipv4
  mdt default 232.1.1.1
 exit-address-family
!
router bgp 1000
 address-family ipv4 mdt
  neighbor 10.255.255.3 activate
  neighbor 10.255.255.3 send-community extended
  neighbor 10.255.255.4 activate
  neighbor 10.255.255.4 send-community extended
 exit-address-family
!
```

```
interface Loopback0
 ip address 10.255.255.6 255.255.255.255
 ip pim sparse-mode
!
interface GigabitEthernet1
 vrf forwarding Customer1
 ip address 192.168.68.6 255.255.255.0
 ip pim sparse-mode
!
interface GigabitEthernet7
 ip address 10.2.6.6 255.255.255.0
 ip pim sparse-mode
!
interface GigabitEthernet8
 ip address 10.1.6.6 255.255.255.0
 ip pim sparse-mode
!
```

These configuration was only on IOS-XE devices, Let's configure the IOS-XR device as well:

XRv9k-3:

```
multicast-routing
 address-family ipv4
  interface Loopback0
   enable
  !
  interface GigabitEthernet0/0/0/4
   enable
  !
  interface GigabitEthernet0/0/0/6
   enable
  !
  mdt source Loopback0
 !
 vrf Customer1
  address-family ipv4
```

```
   interface GigabitEthernet0/0/0/3
    enable
   !
   mdt source Loopback0
   mdt default ipv4 232.1.1.1
  !
 !
!
route-policy RPF_Customer1
  set core-tree pim-default
end-policy
router pim
 address-family ipv4
  rpf topology route-policy RPF_Customer1
  interface Loopback0
   enable
  !
  interface GigabitEthernet0/0/0/4
   enable
  !
  interface GigabitEthernet0/0/0/6
   enable
  !
 !
 vrf Customer1
  address-family ipv4
   rpf topology route-policy RPF_Customer1
   interface GigabitEthernet0/0/0/3
    enable
   !
  !
 !
!
```

```
router bgp 1000
 bgp log neighbor changes detail
 address-family vpnv4 unicast
 !
 address-family ipv4 mdt
 !
 neighbor 10.255.255.4
  !
   address-family ipv4 mdt
  !
  neighbor 10.255.255.6
  !
   address-family ipv4 mdt
  !
 !
!
commit
```

The same configuration but in IOS-XR language!

One more step is needed on IOS-XE PE devices, take a look at the mdt multicast group: 232.1.1.1:

It is an SSM range (Source Specific Multicast). By default IOS-XE does not consider this range as an SSM range, we need to manually enable it:

```
CSR1k-4:
ip pim ssm default
CSR1k-6:
ip pim ssm default
```

Last step is to enable multicast routing on the core devices:

```
CSR1k-1:
ip pim ssm default
ip multicast-routing distributed
interface GigabitEthernet1
 ip pim sparse-mode
```

```
!
interface GigabitEthernet5
 ip pim sparse-mode
!
interface GigabitEthernet6
 ip pim sparse-mode
!
interface GigabitEthernet8
 ip pim sparse-mode
!
```

CSR1k-2:

```
ip pim ssm default
ip multicast-routing distributed
interface GigabitEthernet1
 ip pim sparse-mode
!
interface GigabitEthernet4
 ip pim sparse-mode
!
interface GigabitEthernet6
 ip pim sparse-mode
!
interface GigabitEthernet7
 ip pim sparse-mode
!
```

Verification:

Let's check the core device to see what we have in the Multicast routing table. We will see some (S, G) entries in the output, those (S, G) will be (PE Loopback 0 IP address, 232.1.1.1), because each of those routers are going to join that group (they have a customer device connected to them), and they want to receive and also forward the multicast traffic to the other peers (Remote PE devices).

```
CSR1k-2#show ip mroute | begin Out
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(10.255.255.3, 232.1.1.1), 01:28:22/00:02:49, flags: sT
  Incoming interface: GigabitEthernet4, RPF nbr 10.2.3.3
  Outgoing interface list:
    GigabitEthernet6, Forward/Sparse, 01:28:22/00:02:43
    GigabitEthernet7, Forward/Sparse, 01:28:22/00:02:49

(10.255.255.6, 232.1.1.1), 01:28:22/00:03:06, flags: sT
  Incoming interface: GigabitEthernet7, RPF nbr 10.2.6.6
  Outgoing interface list:
    GigabitEthernet6, Forward/Sparse, 01:28:22/00:02:43
    GigabitEthernet4, Forward/Sparse, 01:28:22/00:03:06

(10.255.255.4, 232.1.1.1), 01:28:22/00:03:06, flags: sT
  Incoming interface: GigabitEthernet6, RPF nbr 10.2.4.4
  Outgoing interface list:
    GigabitEthernet4, Forward/Sparse, 01:28:22/00:03:06
    GigabitEthernet7, Forward/Sparse, 01:28:22/00:02:40

(*, 224.0.1.40), 01:53:17/00:02:40, RP 0.0.0.0, flags: DCL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    GigabitEthernet1, Forward/Sparse, 01:53:17/00:02:40
```

How these PE devices discovered each other? How they know which PE device needs to receive the multicast traffic destined to 232.1.1.1?

MP-BGP IPv4 MDT is the answer!

Let's check the BGP neghborship and also it's detailed information (routes detailed information):

```
CSR1k-4#show bgp ipv4 mdt all summary  | begin Ne
Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
10.255.255.3    4         1000      98     109        4    0    0 01:31:33        1
10.255.255.6    4         1000     110     114        4    0    0 01:32:24        1

CSR1k-4#show bgp ipv4 mdt all  | begin Net
     Network          Next Hop            Metric LocPrf Weight Path
Route Distinguisher: 1000:1001 (default for vrf Customer1)
 *>i  10.255.255.3/32  10.255.255.3                 100      0 i
 *>   10.255.255.4/32  0.0.0.0                              0 ?
 *>i  10.255.255.6/32  10.255.255.6             0    100      0 ?

CSR1k-4#show bgp ipv4 mdt all  10.255.255.3
BGP routing table entry for 1000:1001:10.255.255.3/32    version 4
Paths: (1 available, best #1, table IPv4-MDT-BGP-Table)
  Not advertised to any peer
  Refresh Epoch 1
  Local
    10.255.255.3 from 10.255.255.3 (10.255.255.3)
      Origin IGP, localpref 100, valid, internal, best,
      MDT group address: 232.1.1.1

      rx pathid: 0, tx pathid: 0x0
      Updated on Apr 11 2021 21:36:41 UTC
```

Everything seems to be fine, let's stream a video on the server, and on the clients join those multicast groups to get that streamed video:

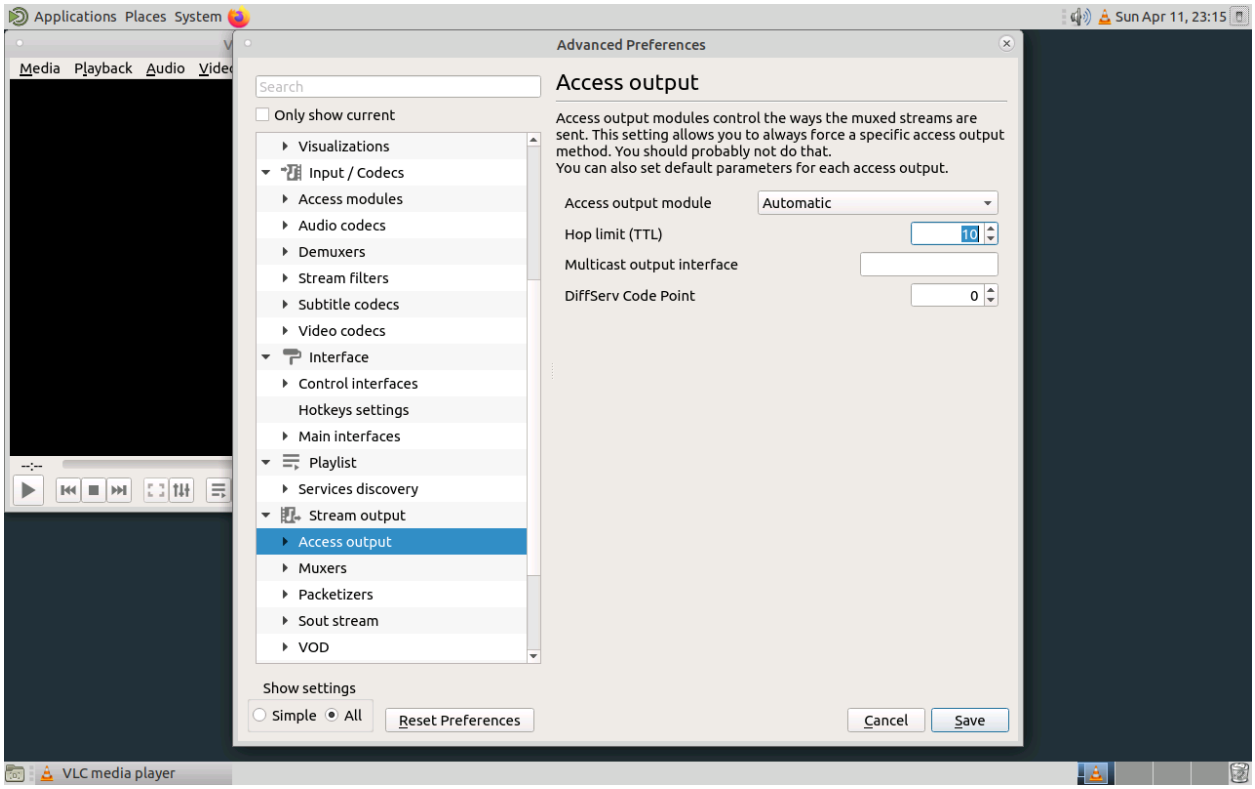
For this purpose we will use VLC media Player:

First of all let's make sure that Stream data IP packet TTL is set to something higher than 1, otherwise the packets will be dropped on the First-Hop router:

In order to do that:

- Open VLC APP
- Click on the Tools
- Go to the Preferences
- Show settings (All)
- Stream Output section

- Access Output
- Set HOP Limit TTL to something greater than for example 10 (it depends on your network topology)



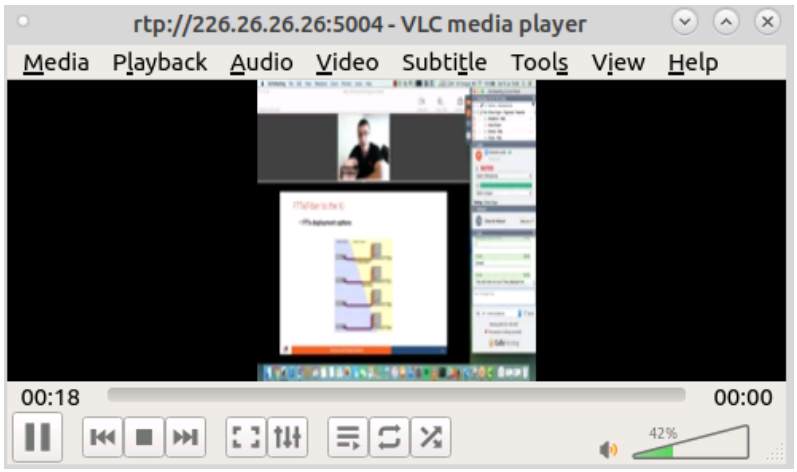
Next step is to stream a video:

- Click on Media
- Stream
- Add
- Choose your multimedia file (video)
- Click on Stream
- Next
- New destination: RTP/MPEG Transport Stream
- Click on Add
- Address: Any multicast group address (as an example 226.26.26.26)
- Base Port: Any port that you want (UDP Port number) the default is 5004
- Next
- Profile: Video for android SD Low
- Next
- Stream

That is all you need to do on the server side

On the client side:

- Open VLC Media Player
- Click on Media Tab
- Open Network Stream
- URL: rtp://@226.26.26.26:5004
- Play



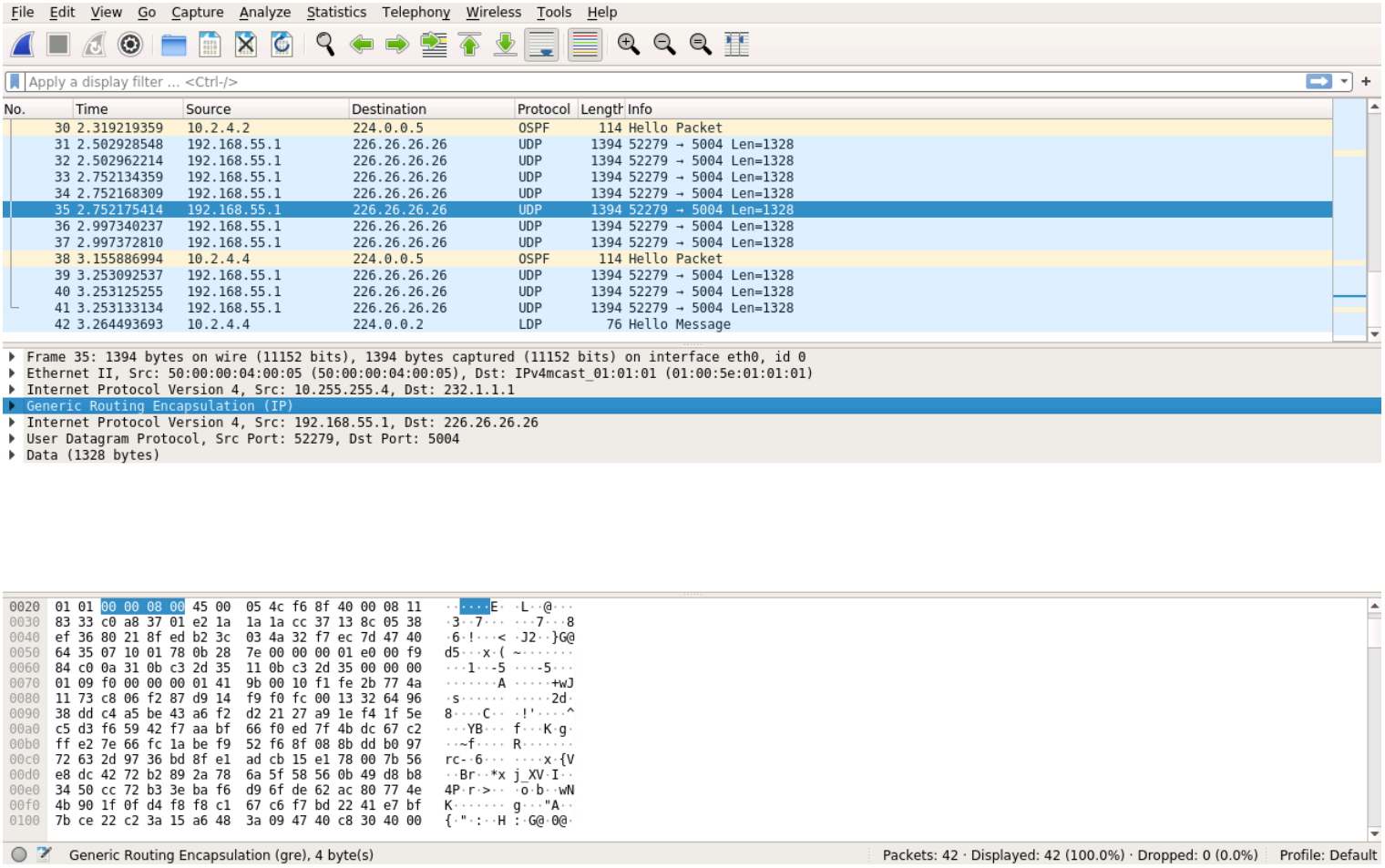

```
CSR1k-4#show ip mroute vrf Customer1 | begin Out
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 226.26.26.26), 00:02:40/stopped, RP 10.255.255.8, flags: SP
  Incoming interface: Tunnel0, RPF nbr 10.255.255.6
  Outgoing interface list: Null

(192.168.55.1, 226.26.26.26), 00:02:40/00:00:19, flags: T
  Incoming interface: GigabitEthernet3, RPF nbr 192.168.45.5
  Outgoing interface list:
    Tunnel0, Forward/Sparse, 00:02:40/00:03:23

(*, 224.0.1.40), 02:14:10/00:02:52, RP 0.0.0.0, flags: DPL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list: Null
```

As you can realize from the above output, The Tunnel interface is the outgoing interface of this multicast group traffic, the source is 192.168.55.1 (Server) and the group is 226.26.26.26.

Thanks to the PIM as the Signaling protocol, other PE devices can get all these information from othe PE devices:

```
CSR1k-6#show ip pim vrf Customer1 neighbor
PIM Neighbor Table
Mode: B - Bidir Capable, DR - Designated Router, N - Default DR Priority,
      P - Proxy Capable, S - State Refresh Capable, G - GenID Capable,
      L - DR Load-balancing Capable
Neighbor          Interface                Uptime/Expires    Ver   DR
Address                                                            Prio/Mode
192.168.68.8      GigabitEthernet1         01:59:03/00:01:42 v2    1 / DR S P G
10.255.255.3      Tunnel1                  01:52:15/00:01:18 v2    1 / G
10.255.255.4      Tunnel1                  01:52:15/00:01:15 v2    1 / S P G

CSR1k-6#show ip mroute vrf Customer1 | begin Out
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 226.26.26.26), 00:06:17/00:02:47, RP 10.255.255.8, flags: S
  Incoming interface: GigabitEthernet1, RPF nbr 192.168.68.8
  Outgoing interface list:
    Tunnel1, Forward/Sparse, 00:05:40/00:02:47

(192.168.55.1, 226.26.26.26), 00:06:17/00:03:01, flags: T
  Incoming interface: Tunnel1, RPF nbr 10.255.255.4
  Outgoing interface list:
    GigabitEthernet1, Forward/Sparse, 00:06:17/00:02:31

(*, 224.0.1.40), 01:54:01/00:02:15, RP 0.0.0.0, flags: DPL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list: Null
```

Let's capture those streams on the physical link too see what is happening behind the scenes:



The actual data is encapsulated inside the GRE header.

- Actual data from the customer device: 192.168.55.1 to the destination of 226.26.26.26
- The outer IP header IP address: from 10.255.255.4 (CSR1K-4 Loopback 0 IP) to the destination group of 232.1.1.1

Task 07:

- Configure Multicast VPN using Profile 1
  Default MDT – MLDP MP2MP PIM C-Mcast Signaling

Solution:

In all of these profiles we are using Default MDT, because this is the only option that is supported on these virtual platforms.

The XRv9k device (IOS-XR device) at the moment of the writing of this workbook only supports GRE as the encapsulation. MPLS encapsulation is not supported. So We will only configure the profiles that have MLDP (Multipoint LDP) on IOS-XE platform (CSR1000v).

In this profile:

- Default MDT is going to be used to form Multicast Distribution Tree between PE devices
- MLDP (Multipoint or Multicast LDP) is going to be used in order to encapsulate the Customer Multicast traffic (This device only supports MP2MP
- PIM is going to be used as customer multicast routes signaling (*, G), (S, G) etc… .

The only differences between the previous task and this task are:

- There is no BGP Auto-Discovery in this option, MLDP is going to handle this
- There is no GRE encapsulation
- There is no need to enable Multicast Routing in the core of the SP, We need to only enable Multicast Routing under VRFs for the customer facing interfaces

First Erase the configuration on all of the Service Provider IOS-XE devices (All PEs and Core devices):

```
CSR1k-4, CSR1k-1, CSR1k-2, CSR1k-6:
do write erase
```

Then copy paste these Configuration:

```
CSR1k-1:
hostname CSR1k-1
!
no ip icmp rate-limit unreachable
!
no ip domain lookup
!
```

```
mpls label range 1000 1999
!
interface Loopback0
 ip address 10.255.255.1 255.255.255.255
 ip ospf 1000 area 0
!
interface Loopback12
 ip address 12.12.12.12 255.255.255.255
 ip ospf 1000 area 0
!
interface GigabitEthernet1
 bandwidth 10000000
 ip address 10.1.2.1 255.255.255.0
 ip ospf network point-to-point
 ip ospf 1000 area 0
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet2
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet3
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet4
```

```
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet5
 ip address 10.1.4.1 255.255.255.0
 ip ospf network point-to-point
 ip ospf 1000 area 0
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet6
 ip address 10.1.3.1 255.255.255.0
 ip ospf network point-to-point
 ip ospf 1000 area 0
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet7
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet8
 ip address 10.1.6.1 255.255.255.0
 ip ospf network point-to-point
 ip ospf 1000 area 0
 negotiation auto
```

```
 no mop enabled
 no mop sysid
!
interface GigabitEthernet9
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet10
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet11
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
router ospf 1000
 router-id 0.0.0.1
 auto-cost reference-bandwidth 100000
 passive-interface Loopback0
 mpls ldp autoconfig
!
router ospf 1
!
ip forward-protocol nd
!
```

```
!
!
!
!
!
mpls ldp router-id Loopback0
!
!
!
control-plane
!
!
!
!
!
!
line con 0
 exec-timeout 0 0
 logging synchronous
 length 26
 width 123
 stopbits 1
line vty 0 4
 login
 transport input ssh
!
!
```

CSR1K-2:

```
hostname CSR1k-2
!
no ip icmp rate-limit unreachable
!
no ip domain lookup
```

```
!
mpls label range 2000 2999
!
interface Loopback0
 ip address 10.255.255.2 255.255.255.255
 ip ospf 1000 area 0
!
interface Loopback12
 ip address 12.12.12.12 255.255.255.255
 ip ospf 1000 area 0
!
interface GigabitEthernet1
 bandwidth 10000000
 ip address 10.1.2.2 255.255.255.0
 ip ospf network point-to-point
 ip ospf 1000 area 0
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet2
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet3
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
```

```
interface GigabitEthernet4
 ip address 10.2.3.4 255.255.255.0
 ip ospf network point-to-point
 ip ospf 1000 area 0
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet5
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet6
 ip address 10.2.4.2 255.255.255.0
 ip ospf network point-to-point
 ip ospf 1000 area 0
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet7
 ip address 10.2.6.2 255.255.255.0
 ip ospf network point-to-point
 ip ospf 1000 area 0
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet8
 no ip address
 shutdown
```

```
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet9
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet10
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet11
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
router ospf 1000
 router-id 0.0.0.2
 auto-cost reference-bandwidth 100000
 passive-interface Loopback0
 mpls ldp autoconfig
!
ip forward-protocol nd
!
mpls ldp router-id Loopback0
```

```
!
line con 0
 exec-timeout 0 0
 logging synchronous
 length 26
 width 123
 stopbits 1
line vty 0 4
 login
 transport input ssh
!
```

The core devices had simple configuration, not additional configuration other that basic IGP and MPLS LDP is needed.

Let's configure PE devices:

CSR1k-4:

```
hostname CSR1k-4
!
vrf definition Customer1
 rd 1000:1001
 vpn id 1000:1001
 !
 address-family ipv4
  mdt default mpls mldp 12.12.12.12
  route-target export 1000:1001
  route-target import 1000:1001
 exit-address-family
!
!
no ip icmp rate-limit unreachable
!
ip multicast-routing vrf Customer1 distributed
!
mpls label range 4000 4999
mpls mldp logging notifications
```

```
!
interface Loopback0
 ip address 10.255.255.4 255.255.255.255
 ip ospf 1000 area 0
!
interface GigabitEthernet1
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet2
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet3
 vrf forwarding Customer1
 ip address 192.168.45.4 255.255.255.0
 ip pim sparse-mode
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet4
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
```

```
interface GigabitEthernet5
 ip address 10.1.4.4 255.255.255.0
 ip ospf network point-to-point
 ip ospf 1000 area 0
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet6
 ip address 10.2.4.4 255.255.255.0
 ip ospf network point-to-point
 ip ospf 1000 area 0
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet7
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet8
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet9
 no ip address
 shutdown
 negotiation auto
```

```
 no mop enabled
 no mop sysid
!
interface GigabitEthernet10
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet11
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
router ospf 1000
 router-id 0.0.0.4
 auto-cost reference-bandwidth 100000
 passive-interface Loopback0
 mpls ldp autoconfig
!
router bgp 1000
 bgp log-neighbor-changes
 no bgp default ipv4-unicast
 neighbor 10.255.255.3 remote-as 1000
 neighbor 10.255.255.3 update-source Loopback0
 neighbor 10.255.255.6 remote-as 1000
 neighbor 10.255.255.6 update-source Loopback0
 !
 address-family ipv4
 exit-address-family
 !
```

```
 address-family vpnv4
  neighbor 10.255.255.3 activate
  neighbor 10.255.255.3 send-community extended
  neighbor 10.255.255.6 activate
  neighbor 10.255.255.6 send-community extended
 exit-address-family
 !
 address-family ipv4 vrf Customer1
  redistribute connected metric 20000
  neighbor 192.168.45.5 remote-as 65001
  neighbor 192.168.45.5 activate
  neighbor 192.168.45.5 as-override
 exit-address-family
!
mpls ldp router-id Loopback0
!
!
line con 0
 exec-timeout 0 0
 logging synchronous
 length 26
 width 123
 stopbits 1
line vty 0 4
 login
 transport input ssh
!
```

You can realize that, Profile 1 configuration is very easy and straight forward.

The only commands that you need to configure are highlighted on the above configuration sections.

If you take a closer look at the core deevices, we have defined Loopback 12 on both of them (IP address: 12.12.12.12 .Those two devices are going to be used as the MLDP root nodes:

Let’s configure the other PE device, then we will discuss in detail about how MLDP works:

CSR1K-6:

```
hostname CSR1k-6
!
vrf definition Customer1
 rd 1000:1001
 vpn id 1000:1001
 !
 address-family ipv4
  mdt default mpls mldp 12.12.12.12
  route-target export 1000:1001
  route-target import 1000:1001
 exit-address-family
!
ip multicast-routing vrf Customer1 distributed
!
mpls label range 6000 6999
mpls mldp logging notifications
!
interface Loopback0
 ip address 10.255.255.6 255.255.255.255
 ip ospf 1000 area 0
!
interface GigabitEthernet1
 vrf forwarding Customer1
 ip address 192.168.68.6 255.255.255.0
 ip pim sparse-mode
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet2
 no ip address
 shutdown
 negotiation auto
 no mop enabled
```

```
 no mop sysid
!
interface GigabitEthernet3
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet4
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet5
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet6
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet7
 ip address 10.2.6.6 255.255.255.0
 ip ospf network point-to-point
 ip ospf 1000 area 0
```

```
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet8
 ip address 10.1.6.6 255.255.255.0
 ip ospf network point-to-point
 ip ospf 1000 area 0
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet9
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet10
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet11
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
router ospf 1000
```

```
 router-id 0.0.0.6
 auto-cost reference-bandwidth 100000
 passive-interface Loopback0
 mpls ldp autoconfig
!
router bgp 1000
 bgp log-neighbor-changes
 no bgp default ipv4-unicast
 neighbor 10.255.255.3 remote-as 1000
 neighbor 10.255.255.3 update-source Loopback0
 neighbor 10.255.255.4 remote-as 1000
 neighbor 10.255.255.4 update-source Loopback0
 !
 address-family ipv4
 exit-address-family
 !
 address-family vpnv4
  neighbor 10.255.255.3 activate
  neighbor 10.255.255.3 send-community extended
  neighbor 10.255.255.4 activate
  neighbor 10.255.255.4 send-community extended
 exit-address-family
 !
 address-family ipv4 vrf Customer1
  redistribute connected metric 20000
  neighbor 192.168.68.8 remote-as 65001
  neighbor 192.168.68.8 activate
  neighbor 192.168.68.8 as-override
 exit-address-family
!
ip forward-protocol nd
!
mpls ldp router-id Loopback0
!
```

```
line con 0
 exec-timeout 0 0
 logging synchronous
 length 26
 width 123
 stopbits 1
line vty 0 4
 login
 transport input ssh
!
```

Verification:

```
CSR1k-6#show run vrf
Building configuration...

Current configuration : 597 bytes
vrf definition Customer1
 rd 1000:1001
 vpn id 1000:1001
 !
 address-family ipv4
  mdt default mpls mldp 12.12.12.12
--more–

CSR1k-6#show mpls mldp root

 Root node    : 12.12.12.12
  Metric      : 101
  Distance    : 110
  Interface   : GigabitEthernet7 (via unicast RT)
  FEC count   : 1
  Path count  : 2
  Path(s)     : 10.2.6.2         LDP nbr: 10.255.255.2:0    GigabitEthernet7
              : 10.1.6.1         LDP nbr: 10.255.255.1:0    GigabitEthernet8
```

First of all 12.12.12.12 is the Root node.

```
CSR1k-6#show mpls mldp neighbors


 MLDP peer ID    : 10.255.255.2:0, uptime 03:10:36 Up,
  Target Adj     : No
  Session hndl   : 1
  Upstream count : 1
  Branch count   : 0
  Path count     : 1
  Path(s)        : 10.2.6.2          LDP GigabitEthernet7
  Nhop count     : 1
  Nhop list      : 10.2.6.2


 MLDP peer ID    : 10.255.255.1:0, uptime 03:10:36 Up,
  Target Adj     : No
  Session hndl   : 2
  Upstream count : 0
  Branch count   : 0
  Path count     : 1
  Path(s)        : 10.1.6.1          LDP GigabitEthernet8
  Nhop count     : 1
  Nhop list      : 10.1.6.1
```

There are two MLDP neighbors (Core routers).

```
CSR1k-6#show mpls mldp bindings
System ID: 3
Type: MP2MP, Root Node: 12.12.12.12, Opaque Len: 14
Opaque value: [mdt 1000:1001 0]
lsr: 10.255.255.2:0, remote binding[U]: 2009, local binding[D]: 6013 active
```

There are some remote and local labels.

Why we need these Labels?

Let's take a look at the root node detailed MLDP detabase to find the answer:

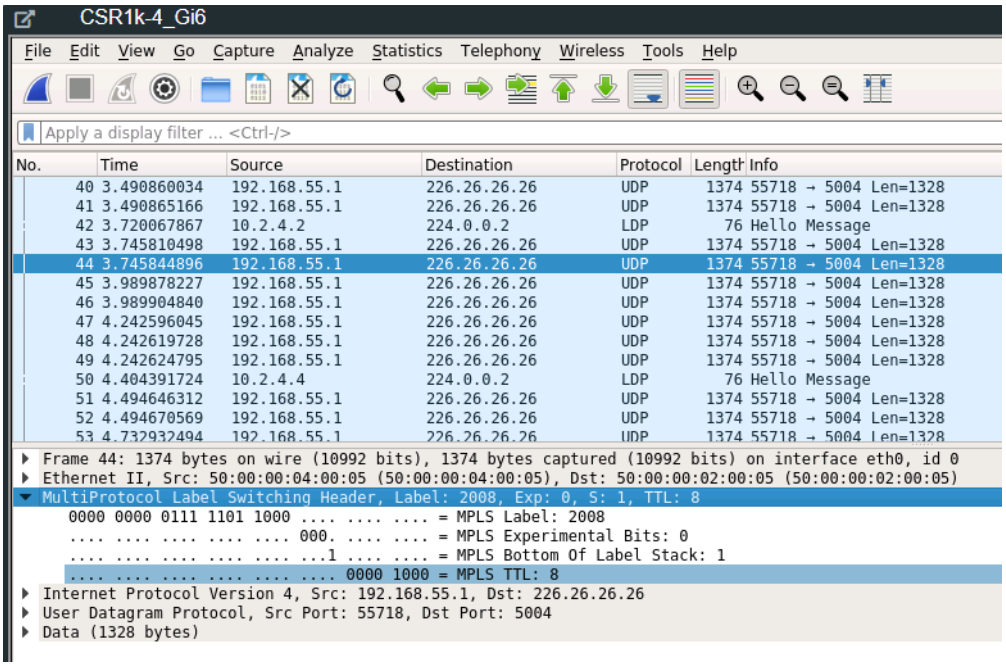
```
CSR1k-2#show mpls mldp database
  * For interface indicates MLDP recursive forwarding is enabled
  * For RPF-ID indicates wildcard value
  > Indicates it is a Primary MLDP MDT Branch

LSM ID : 1   Type: MP2MP   Uptime : 00:35:06
  FEC Root           : 12.12.12.12 (we are the root)
  Opaque decoded     : [mdt 1000:1001 0]
  Opaque length      : 11 bytes
  Opaque value       : 02 000B 00100000001001000000000
  Upstream client(s) :
    None
      Expires        : N/A           Path Set ID  : 1
  Replication client(s):
    10.255.255.4:0
      Uptime         : 00:35:06      Path Set ID  : 2
      Out label (D)  : 4013          Interface    : GigabitEthernet6*
      Local label (U): 2008          Next Hop     : 10.2.4.4
    10.255.255.6:0
      Uptime         : 00:35:02      Path Set ID  : 3
      Out label (D)  : 6013          Interface    : GigabitEthernet7*
      Local label (U): 2009          Next Hop     : 10.2.6.6
```

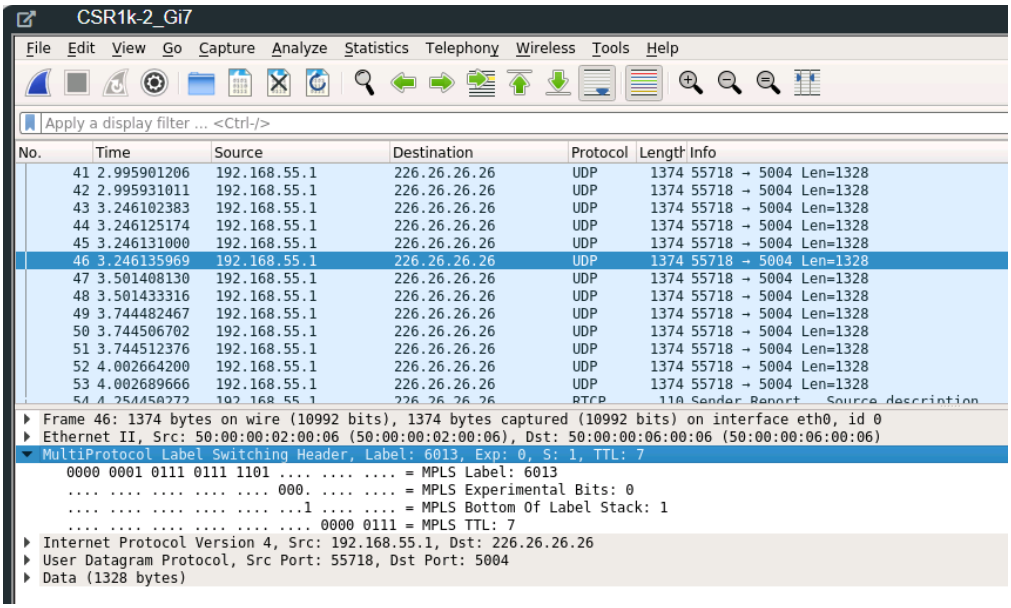
As you can realize from the above output MDT data is encoded inside MLDP messages (Opaque value), the router decodes it and can find out about the MDT and how to build that tree.

There are also Local and Outgoing/Outbound labels.

- The root node advertises it's local MLDP labels to the other neighbors
- Each of the PE devices advertise their own Local label to the Root node
- When a multicast data comes from the Customer device (Let's say CSR1k-5), it needs to be sent to the CSR1k-6
- CSR1k-4 takes a look at the Out Label which is 2008 (received from Root node) and forwards the packet towards the root node
- Root node will receive it, it takes a look at the MLDP label binding and swaps label 2008 with 6013 (because CSR1k-6 advertised him that label before)
- There is no PHP in MLDP

Let's stream a video and capture the data to see what is happening behind the scenes:





That easy ☺

Task 08:

- Configure Multicast VPN Using Profile 3
  Default MDT – GRE – BGP-AD – PIM C-Mcast Signaling

Solution:

In this profile:

- Default MDT (We already discussed about it in the previous tasks)
- GRE: As the Customer Multicast traffic encapsulation
- BGP-AD: We will use MP-BGP IPv4 mVPN AF/SAF for the Auto-Discovery purpose (PE devices can find about each other using BGP)
- PIM: (We already discussed about it in the previous tasks)

This is just like Profile 0, but instead of using IPv4 MDT, we will use IPv4 mVPN.

First of all let's erase all SP routers configurations:

```
CSR1k-4, CSR1k-1, CSR1k-2, CSR1k-6:
do write erase

XRv9k-3:
do commit replace
```

And copy paste all these configurations:

```
CSR1k-1:
hostname CSR1k-1
!
ip multicast-routing distributed
!
no ip domain lookup
!
mpls label range 1000 1999
!
interface Loopback0
 ip address 10.255.255.1 255.255.255.255
 ip ospf 1000 area 0
!
```

```
interface GigabitEthernet1
 bandwidth 10000000
 ip address 10.1.2.1 255.255.255.0
 ip pim sparse-mode
 ip ospf network point-to-point
 ip ospf 1000 area 0
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet2
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet3
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet4
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet5
 ip address 10.1.4.1 255.255.255.0
 ip pim sparse-mode
```

```
 ip ospf network point-to-point
 ip ospf 1000 area 0
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet6
 ip address 10.1.3.1 255.255.255.0
 ip ospf network point-to-point
 ip ospf 1000 area 0
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet7
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet8
 ip address 10.1.6.1 255.255.255.0
 ip pim sparse-mode
 ip ospf network point-to-point
 ip ospf 1000 area 0
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet9
 no ip address
 shutdown
 negotiation auto
```

```
 no mop enabled
 no mop sysid
!
interface GigabitEthernet10
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet11
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
router ospf 1000
 router-id 0.0.0.1
 auto-cost reference-bandwidth 100000
 passive-interface Loopback0
 mpls ldp autoconfig
!
ip pim ssm default
!
mpls ldp router-id Loopback0
!
```

CSR1k-2:

```
hostname CSR1k-2
!
ip multicast-routing distributed
!
no ip domain lookup
```

```
!
mpls label range 2000 2999
!
interface Loopback0
 ip address 10.255.255.2 255.255.255.255
 ip ospf 1000 area 0
!
interface GigabitEthernet1
 bandwidth 10000000
 ip address 10.1.2.2 255.255.255.0
 ip pim sparse-mode
 ip ospf network point-to-point
 ip ospf 1000 area 0
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet2
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet3
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet4
 ip address 10.2.3.4 255.255.255.0
 ip pim sparse-mode
```

```
 ip ospf network point-to-point
 ip ospf 1000 area 0
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet5
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet6
 ip address 10.2.4.2 255.255.255.0
 ip pim sparse-mode
 ip ospf network point-to-point
 ip ospf 1000 area 0
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet7
 ip address 10.2.6.2 255.255.255.0
 ip pim sparse-mode
 ip ospf network point-to-point
 ip ospf 1000 area 0
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet8
 no ip address
 shutdown
```

```
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet9
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet10
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet11
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
router ospf 1000
 router-id 0.0.0.2
 auto-cost reference-bandwidth 100000
 passive-interface Loopback0
 mpls ldp autoconfig
!
ip forward-protocol nd
ip pim ssm default
!
```

```
mpls ldp router-id Loopback0
!


CSR1k-4:
hostname CSR1k-4
!
vrf definition Customer1
 rd 1000:1001
 !
 address-family ipv4
  mdt auto-discovery pim
  mdt default 232.1.1.1
  route-target export 1000:1001
  route-target import 1000:1001
 exit-address-family
!
ip multicast-routing distributed
ip multicast-routing vrf Customer1 distributed
!
no ip domain lookup
!
mpls label range 4000 4999
!
interface Loopback0
 ip address 10.255.255.4 255.255.255.255
 ip pim sparse-mode
 ip ospf 1000 area 0
!
interface GigabitEthernet1
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
```

```
interface GigabitEthernet2
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet3
 vrf forwarding Customer1
 ip address 192.168.45.4 255.255.255.0
 ip pim sparse-mode
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet4
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet5
 ip address 10.1.4.4 255.255.255.0
 ip pim sparse-mode
 ip ospf network point-to-point
 ip ospf 1000 area 0
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet6
 ip address 10.2.4.4 255.255.255.0
 ip pim sparse-mode
```

```
 ip ospf network point-to-point
 ip ospf 1000 area 0
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet7
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet8
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet9
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet10
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
```

```
interface GigabitEthernet11
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
router ospf 1000
 router-id 0.0.0.4
 auto-cost reference-bandwidth 100000
 passive-interface Loopback0
 mpls ldp autoconfig
!
router bgp 1000
 bgp log-neighbor-changes
 no bgp default ipv4-unicast
 neighbor 10.255.255.3 remote-as 1000
 neighbor 10.255.255.3 update-source Loopback0
 neighbor 10.255.255.6 remote-as 1000
 neighbor 10.255.255.6 update-source Loopback0
 !
 address-family ipv4
 exit-address-family
 !
 address-family ipv4 mvpn
  neighbor 10.255.255.3 activate
  neighbor 10.255.255.3 send-community extended
  neighbor 10.255.255.6 activate
  neighbor 10.255.255.6 send-community extended
 exit-address-family
 !
 address-family vpnv4
  neighbor 10.255.255.3 activate
  neighbor 10.255.255.3 send-community extended
```

```
  neighbor 10.255.255.6 activate
  neighbor 10.255.255.6 send-community extended
 exit-address-family
 !
 address-family ipv4 vrf Customer1
  redistribute connected metric 20000
  neighbor 192.168.45.5 remote-as 65001
  neighbor 192.168.45.5 activate
  neighbor 192.168.45.5 as-override
 exit-address-family
!
ip forward-protocol nd
ip pim ssm default
!
mpls ldp router-id Loopback0
!
```

The configurations that are important are highlighted in the above command boxes. As you realized we have enabled a new address family for Auto Discovery Purpose (IPv4 mVPN address family) for the neighbors (PE devices). Also we defined mdt auto-discovery pim command under the VRF definition ipv4 unicast address family.

mdt auto-discovery means we are using BGP as the auto-discovery, and the pim at the end of the command means we are using pim as the signaling protocol.

Let's configure the other two PE devices:

CSR1k-6:

```
hostname CSR1k-6
!
vrf definition Customer1
 rd 1000:1001
 !
 address-family ipv4
  mdt auto-discovery pim
  mdt default 232.1.1.1
  route-target export 1000:1001
```

```
  route-target import 1000:1001
 exit-address-family
!
no ip icmp rate-limit unreachable
!
ip multicast-routing distributed
ip multicast-routing vrf Customer1 distributed
!
no ip domain lookup
!
mpls label range 6000 6999
!
interface Loopback0
 ip address 10.255.255.6 255.255.255.255
 ip pim sparse-mode
 ip ospf 1000 area 0
!
interface GigabitEthernet1
 vrf forwarding Customer1
 ip address 192.168.68.6 255.255.255.0
 ip pim sparse-mode
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet2
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet3
 no ip address
```

```
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet4
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet5
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet6
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet7
 ip address 10.2.6.6 255.255.255.0
 ip pim sparse-mode
 ip ospf network point-to-point
 ip ospf 1000 area 0
 negotiation auto
 no mop enabled
 no mop sysid
```

```
!
interface GigabitEthernet8
 ip address 10.1.6.6 255.255.255.0
 ip pim sparse-mode
 ip ospf network point-to-point
 ip ospf 1000 area 0
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet9
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet10
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet11
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
router ospf 1000
 router-id 0.0.0.6
 auto-cost reference-bandwidth 100000
```

```
 passive-interface Loopback0
 mpls ldp autoconfig
!
router bgp 1000
 bgp log-neighbor-changes
 no bgp default ipv4-unicast
 neighbor 10.255.255.3 remote-as 1000
 neighbor 10.255.255.3 update-source Loopback0
 neighbor 10.255.255.4 remote-as 1000
 neighbor 10.255.255.4 update-source Loopback0
 !
 address-family ipv4
 exit-address-family
 !
 address-family ipv4 mvpn
  neighbor 10.255.255.3 activate
  neighbor 10.255.255.3 send-community extended
  neighbor 10.255.255.4 activate
  neighbor 10.255.255.4 send-community extended
 exit-address-family
 !
 address-family vpnv4
  neighbor 10.255.255.3 activate
  neighbor 10.255.255.3 send-community extended
  neighbor 10.255.255.4 activate
  neighbor 10.255.255.4 send-community extended
 exit-address-family
 !
 address-family ipv4 vrf Customer1
  redistribute connected metric 20000
  neighbor 192.168.68.8 remote-as 65001
  neighbor 192.168.68.8 activate
  neighbor 192.168.68.8 as-override
 exit-address-family
```

```
!
ip forward-protocol nd
ip pim ssm default
!
mpls ldp router-id Loopback0
!
```

The last step is to configure IOS-XR device, the logic of the implementation is the same but with different kind of configuration language!

XRv9k-3:

```
hostname XRv9k-3
logging console debugging
domain lookup disable
!
vrf Customer1
 address-family ipv4 unicast
  import route-target
   1000:1001
  !
  export route-target
   1000:1001
  !
 !
!
line console
 exec-timeout 0 0
 width 123
 length 26
!
interface Loopback0
 ipv4 address 10.255.255.3 255.255.255.255
!
interface GigabitEthernet0/0/0/3
 vrf Customer1
```

```
 ipv4 address 192.168.37.3 255.255.255.0
!
interface GigabitEthernet0/0/0/4
 ipv4 address 10.2.3.3 255.255.255.0
!
interface GigabitEthernet0/0/0/5
 shutdown
!
interface GigabitEthernet0/0/0/6
 ipv4 address 10.1.3.3 255.255.255.0
!
route-policy eBGP-PASS
  pass
end-policy
!
route-policy RPF_Customer1
  set core-tree pim-default
end-policy
!
router ospf 1000
 log adjacency changes detail
 router-id 0.0.0.3
 mpls ldp auto-config
 address-family ipv4
 area 0
  interface Loopback0
   passive enable
  !
  interface GigabitEthernet0/0/0/4
   network point-to-point
  !
  interface GigabitEthernet0/0/0/6
   network point-to-point
  !
```

```
 !
!
router bgp 1000
 bgp log neighbor changes detail
 address-family vpnv4 unicast
 !
 address-family ipv4 mvpn
 !
 neighbor 10.255.255.4
  remote-as 1000
  update-source Loopback0
  address-family vpnv4 unicast
  !
  address-family ipv4 mvpn
  !
 !
 neighbor 10.255.255.6
  remote-as 1000
  update-source Loopback0
  address-family vpnv4 unicast
  !
  address-family ipv4 mvpn
  !
 !
 vrf Customer1
  rd 1000:1001
  address-family ipv4 unicast
   redistribute connected metric 20000
  !
  address-family ipv4 mvpn
  !
  neighbor 192.168.37.7
   remote-as 65001
   address-family ipv4 unicast
```

```
    route-policy eBGP-PASS in
    route-policy eBGP-PASS out
    as-override
   !
  !
 !
!
multicast-routing
 address-family ipv4
  interface Loopback0
   enable
  !
  interface GigabitEthernet0/0/0/4
   enable
  !
  interface GigabitEthernet0/0/0/6
   enable
  !
  mdt source Loopback0
 !
 vrf Customer1
  address-family ipv4
   interface GigabitEthernet0/0/0/3
    enable
   !
   mdt source Loopback0
router pim
 address-family ipv4
  interface Loopback0
   enable
  !
  interface GigabitEthernet0/0/0/4
   enable
  !
```

```
  interface GigabitEthernet0/0/0/6
   enable
  !
 !
 vrf Customer1
  address-family ipv4
   rpf topology route-policy RPF_Customer1
   interface GigabitEthernet0/0/0/3
    enable
   !
  !
 !
!
mpls label range table 0 30000 39999
```

Everything is ready and should be working!

Verification:

```
RP/0/RP0/CPU0:XRv9k-3#show bgp ipv4 mvpn summary  | begin Neigh
Mon Apr 12 01:21:41.688 UTC
Neighbor        Spk    AS MsgRcvd MsgSent   TblVer  InQ OutQ  Up/Down  St/PfxRcd
10.255.255.4      0  1000      34      31        8    0    0 00:23:27          1
10.255.255.6      0  1000      33      30        8    0    0 00:23:33          1
RP/0/RP0/CPU0:XRv9k-3#show bgp ipv4 mvpn  | begin Network
Mon Apr 12 01:22:10.788 UTC
   Network            Next Hop            Metric LocPrf Weight Path
Route Distinguisher: 1000:1001 (default for vrf Customer1)
*> [1][10.255.255.3]/40
                      0.0.0.0                                0 i
*>i[1][10.255.255.4]/40
                      10.255.255.4             0    100      0 ?
*>i[1][10.255.255.6]/40
                      10.255.255.6             0    100      0 ?

Processed 3 prefixes, 3 paths
```

We can realize that using the BGP, the PE devices can discover each other:

```
CSR1k-4#show bgp ipv4 mvpn rd 1000:1001 detail


Route Distinguisher: 1000:1001 (default for vrf Customer1)

BGP routing table entry for [1][1000:1001][10.255.255.3]/12, version 13

  Paths: (1 available, best #1, table MVPNv4-BGP-Table, not advertised to EBGP peer)

  Not advertised to any peer

  Refresh Epoch 1

  Local

    10.255.255.3 (metric 201) from 10.255.255.3 (10.255.255.3)

      Origin IGP, localpref 100, valid, internal, best

      Community: no-export

      Extended Community: RT:1000:1001

      PMSI Attribute: Flags: 0x0, Tunnel type: PIM SSM, length 8, label: exp-null, tunnel identifier: <
Source: 10.255.255.3, Group: 232.1.1.1 >

      rx pathid: 0, tx pathid: 0x0

      Updated on Apr 12 2021 00:58:25 UTC

 --more--
```

Task 09:

- Configure Multicast VPN using Profile 11
  Default MDT – GRE – BGP-AD – BGP C-Mcast Signaling

Solution:

Profile 11 is:

- Default MDT (We explained it in the previous tasks)
- GRE (We explained it in the previous tasks)
- BGP-AD for the Auto-Discovery of PE devices
- BGP C-Mcast Signaling: Using BGP instead of PIM for the customer multicast routes signaling (like (*, G), (S, G) entries etc…) .

The configuration of the Profile 11 and the Profile 3 are mostly the same with only a small difference:

NOTE: Make sure that you have already done Task 08, don't erase the configuration of the SP devices.

```
CSR1k-4 and CSR1k-6:
vrf definition Customer1
!
 address-family ipv4
  mdt auto-discovery pim
  mdt default 232.1.1.1
  mdt overlay use-bgp
!
XRv9k-3:
router pim
 !
 vrf Customer1
  address-family ipv4
   mdt c-multicast-routing bgp
   !
  !
 !
!
```

These are the only commands that you need to add to the Profile 3 configuration, then it becomes profile 11!

Let's stream a video and see what happens with BGP routes:

```
RP/0/RP0/CPU0:XRv9k-3#show bgp ipv4 mvpn  | begin Net
Mon Apr 12 01:45:01.813 UTC
   Network            Next Hop            Metric LocPrf Weight Path
Route Distinguisher: 1000:1001 (default for vrf Customer1)
*> [1][10.255.255.3]/40
                      0.0.0.0                                0 i
*>i[1][10.255.255.4]/40
                      10.255.255.4             0    100      0 ?
*>i[1][10.255.255.6]/40
                      10.255.255.6             0    100      0 ?
*>i[5][32][192.168.55.1][32][226.26.26.26]/88
                      10.255.255.4             0    100      0 ?
*> [6][1000:1001][1000][32][10.255.255.8][32][226.26.26.26]/184
                      0.0.0.0                                0 i
*> [7][1000:1001][1000][32][192.168.55.1][32][226.26.26.26]/184
                      0.0.0.0                                0 i
```

There are four types of IPv4 mVPN routes.

Type 1: PMSI Auto-Discovery route

Type 5: Source Active Auto-Discovery route

Type 6: Shared Tree Join Route (Towards RP: 10.255.255.8)

Type 7: Source Tree Join Route (Towards the source: 192.168.55.1)

Task 10:

- Configure Multicast VPN using Profile 13
  Default MDT – MLDP – MP2MP - BGP-AD – BGP C-Mcast Signaling

Solution:

Profile 13 is the almost the same as profile 11, the only difference is the encapsulation.

Instead of usgin mGRE as the encapsulation, it will use MPLS as the encapsulation, so there is no need to enable Multicast Routing in the SP core, the only thing is needed to be enabled is MPLS and MLDP (which is by default enabled in the IOS-XE), you need to only do the Customer VRF configuration in order to get it ready for the multicast forwarding support.

NOTE: XRv9k (IOS-XR) device does not support any profile that is using MLDP. The only encapsulation supported on this platform is GRE. So we skip the configuration of this device.

Let's erase the SP devices and copy paste these configuration:

```
CSR1k-4, CSR1k-1, CSR1k-2, CSR1k-6:
do write erase
```

```
CSR1k-1:
hostname CSR1k-1
!
no ip icmp rate-limit unreachable
!
no ip domain lookup
!
mpls label range 1000 1999
!
interface Loopback0
 ip address 10.255.255.1 255.255.255.255
 ip ospf 1000 area 0
!
interface Loopback12
 ip address 12.12.12.12 255.255.255.255
 ip ospf 1000 area 0
!
```

```
interface GigabitEthernet1
 bandwidth 10000000
 ip address 10.1.2.1 255.255.255.0
 ip pim sparse-mode
 ip ospf network point-to-point
 ip ospf 1000 area 0
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet2
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet3
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet4
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet5
 ip address 10.1.4.1 255.255.255.0
 ip pim sparse-mode
```

```
 ip ospf network point-to-point
 ip ospf 1000 area 0
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet6
 ip address 10.1.3.1 255.255.255.0
 ip ospf network point-to-point
 ip ospf 1000 area 0
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet7
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet8
 ip address 10.1.6.1 255.255.255.0
 ip pim sparse-mode
 ip ospf network point-to-point
 ip ospf 1000 area 0
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet9
 no ip address
 shutdown
 negotiation auto
```

```
 no mop enabled
 no mop sysid
!
interface GigabitEthernet10
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet11
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
router ospf 1000
 router-id 0.0.0.1
 auto-cost reference-bandwidth 100000
 passive-interface Loopback0
 mpls ldp autoconfig
!
mpls ldp router-id Loopback0
!
```

CSR1k-2:

```
hostname CSR1k-2
!
no ip icmp rate-limit unreachable
!
no ip domain lookup
!
mpls label range 2000 2999
```

```
!
interface Loopback0
 ip address 10.255.255.2 255.255.255.255
 ip ospf 1000 area 0
!
interface Loopback12
 ip address 12.12.12.12 255.255.255.255
 ip ospf 1000 area 0
!
interface GigabitEthernet1
 bandwidth 10000000
 ip address 10.1.2.2 255.255.255.0
 ip pim sparse-mode
 ip ospf network point-to-point
 ip ospf 1000 area 0
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet2
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet3
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet4
```

```
 ip address 10.2.3.4 255.255.255.0
 ip pim sparse-mode
 ip ospf network point-to-point
 ip ospf 1000 area 0
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet5
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet6
 ip address 10.2.4.2 255.255.255.0
 ip pim sparse-mode
 ip ospf network point-to-point
 ip ospf 1000 area 0
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet7
 ip address 10.2.6.2 255.255.255.0
 ip pim sparse-mode
 ip ospf network point-to-point
 ip ospf 1000 area 0
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet8
```

```
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet9
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet10
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet11
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
router ospf 1000
 router-id 0.0.0.2
 auto-cost reference-bandwidth 100000
 passive-interface Loopback0
 mpls ldp autoconfig
!
mpls ldp router-id Loopback0
```

CSR1k-4:

```
hostname CSR1k-4
!
vrf definition Customer1
 rd 1000:1001
 vpn id 1000:1001
 !
 address-family ipv4
  mdt auto-discovery mldp
  mdt default mpls mldp 12.12.12.12
  mdt overlay use-bgp
  route-target export 1000:1001
  route-target import 1000:1001
 exit-address-family
!
no ip icmp rate-limit unreachable
!
ip multicast-routing vrf Customer1 distributed
!
!
!
!
!
!
no ip domain lookup
!
subscriber templating
!
mpls label range 4000 4999
mpls mldp logging notifications
multilink bundle-name authenticated
!
interface Loopback0
 ip address 10.255.255.4 255.255.255.255
 ip pim sparse-mode
```

```
 ip ospf 1000 area 0
!
interface GigabitEthernet1
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet2
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet3
 vrf forwarding Customer1
 ip address 192.168.45.4 255.255.255.0
 ip pim sparse-mode
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet4
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet5
 ip address 10.1.4.4 255.255.255.0
 ip pim sparse-mode
```

```
 ip ospf network point-to-point
 ip ospf 1000 area 0
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet6
 ip address 10.2.4.4 255.255.255.0
 ip pim sparse-mode
 ip ospf network point-to-point
 ip ospf 1000 area 0
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet7
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet8
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet9
 no ip address
 shutdown
 negotiation auto
 no mop enabled
```

```
 no mop sysid
!
interface GigabitEthernet10
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet11
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
router ospf 1000
 router-id 0.0.0.4
 auto-cost reference-bandwidth 100000
 passive-interface Loopback0
 mpls ldp autoconfig
!
router bgp 1000
 bgp log-neighbor-changes
 no bgp default ipv4-unicast
 neighbor 10.255.255.3 remote-as 1000
 neighbor 10.255.255.3 update-source Loopback0
 neighbor 10.255.255.6 remote-as 1000
 neighbor 10.255.255.6 update-source Loopback0
 !
 address-family ipv4
 exit-address-family
 !
 address-family ipv4 mvpn
```

```
  neighbor 10.255.255.3 activate
  neighbor 10.255.255.3 send-community extended
  neighbor 10.255.255.6 activate
  neighbor 10.255.255.6 send-community extended
 exit-address-family
 !
 address-family vpnv4
  neighbor 10.255.255.3 activate
  neighbor 10.255.255.3 send-community extended
  neighbor 10.255.255.6 activate
  neighbor 10.255.255.6 send-community extended
 exit-address-family
 !
 address-family ipv4 vrf Customer1
  redistribute connected metric 20000
  neighbor 192.168.45.5 remote-as 65001
  neighbor 192.168.45.5 activate
  neighbor 192.168.45.5 as-override
 exit-address-family
!
mpls ldp router-id Loopback0
!
```

CSR1k-6:

```
hostname CSR1k-6
!
vrf definition Customer1
 rd 1000:1001
 vpn id 1000:1001
 !
 address-family ipv4
  mdt auto-discovery mldp
  mdt default mpls mldp 12.12.12.12
```

```
  mdt overlay use-bgp
  route-target export 1000:1001
  route-target import 1000:1001
 exit-address-family
!
no ip icmp rate-limit unreachable
!
ip multicast-routing vrf Customer1 distributed
!
no ip domain lookup
!
subscriber templating
!
mpls label range 6000 6999
mpls mldp logging notifications
!
interface Loopback0
 ip address 10.255.255.6 255.255.255.255
 ip pim sparse-mode
 ip ospf 1000 area 0
!
interface GigabitEthernet1
 vrf forwarding Customer1
 ip address 192.168.68.6 255.255.255.0
 ip pim sparse-mode
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet2
 no ip address
 shutdown
 negotiation auto
 no mop enabled
```

```
 no mop sysid
!
interface GigabitEthernet3
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet4
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet5
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet6
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet7
 ip address 10.2.6.6 255.255.255.0
 ip pim sparse-mode
 ip ospf network point-to-point
```

```
 ip ospf 1000 area 0
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet8
 ip address 10.1.6.6 255.255.255.0
 ip pim sparse-mode
 ip ospf network point-to-point
 ip ospf 1000 area 0
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet9
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet10
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
!
interface GigabitEthernet11
 no ip address
 shutdown
 negotiation auto
 no mop enabled
 no mop sysid
```

```
router ospf 1000
 router-id 0.0.0.6
 auto-cost reference-bandwidth 100000
 passive-interface Loopback0
 mpls ldp autoconfig
!
router bgp 1000
 bgp log-neighbor-changes
 no bgp default ipv4-unicast
 neighbor 10.255.255.3 remote-as 1000
 neighbor 10.255.255.3 update-source Loopback0
 neighbor 10.255.255.4 remote-as 1000
 neighbor 10.255.255.4 update-source Loopback0
 !
 address-family ipv4
 exit-address-family
 !
 address-family ipv4 mvpn
  neighbor 10.255.255.3 activate
  neighbor 10.255.255.3 send-community extended
  neighbor 10.255.255.4 activate
  neighbor 10.255.255.4 send-community extended
 exit-address-family
 !
 address-family vpnv4
  neighbor 10.255.255.3 activate
  neighbor 10.255.255.3 send-community extended
  neighbor 10.255.255.4 activate
  neighbor 10.255.255.4 send-community extended
 exit-address-family
 !
 address-family ipv4 vrf Customer1
  redistribute connected metric 20000
  neighbor 192.168.68.8 remote-as 65001
  neighbor 192.168.68.8 activate
```

```
  neighbor 192.168.68.8 as-override
 exit-address-family
!
ip forward-protocol nd
!
mpls ldp router-id Loopback0
!
```

It's done! The important configuration commands are highlighted in the command boxes.

```
vrf definition Customer1
 vpn id 1000:1001
 !
 address-family ipv4
  mdt auto-discovery mldp
  mdt default mpls mldp 12.12.12.12
  mdt overlay use-bgp
!
```

- mdt auto-discovery mldp : enables BGP Auto-Discovery for MLDP
- mdt default mpls mldp 12.12.12.12: sets the MPLS tunnel option, and uses MLDP LSP to create default MDT, and also sets the MP2MP LSP root address
- mdt overlay use-bgp: uses the BGP as the overlay signaling

Verification:

```
CSR1k-4#show bgp ipv4 mvpn all  | begin Net
     Network          Next Hop            Metric LocPrf Weight Path
Route Distinguisher: 1000:1001 (default for vrf Customer1)
 *>i  [1][1000:1001][10.255.255.3]/12
                      10.255.255.3                  100      0 i
 *>   [1][1000:1001][10.255.255.4]/12
                      0.0.0.0                            32768 ?
 *>i  [1][1000:1001][10.255.255.6]/12
                      10.255.255.6             0    100      0 ?
 *>   [5][1000:1001][192.168.55.1][226.26.26.26]/18
                      0.0.0.0                            32768 ?
 *>i  [7][1000:1001][1000][192.168.55.1/32][226.26.26.26/32]/22
                      10.255.255.6             0    100      0 ?
```

We have all the information needed in order to tunnel the customer multicast destined data.

```
CSR1k-4#show ip mroute vrf Customer1 | begin Out
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM Join
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 226.26.26.26), 00:01:39/stopped, RP 10.255.255.8, flags: SP
  Incoming interface: Lspvif0, RPF nbr 10.255.255.6
  Outgoing interface list: Null

(192.168.55.1, 226.26.26.26), 00:01:39/00:01:20, flags: TGq
  Incoming interface: GigabitEthernet3, RPF nbr 192.168.45.5
  Outgoing interface list:
    Lspvif0, Forward/Sparse, 00:01:39/00:01:20

(*, 224.0.1.40), 05:12:43/00:02:18, RP 0.0.0.0, flags: DPL
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list: Null
```