An assignment report on

VPLS Lab 4

Submitted for fufillment of award of
Bachelor of Computing and Network Communications
degree

By
Chance Page
26/11/2024

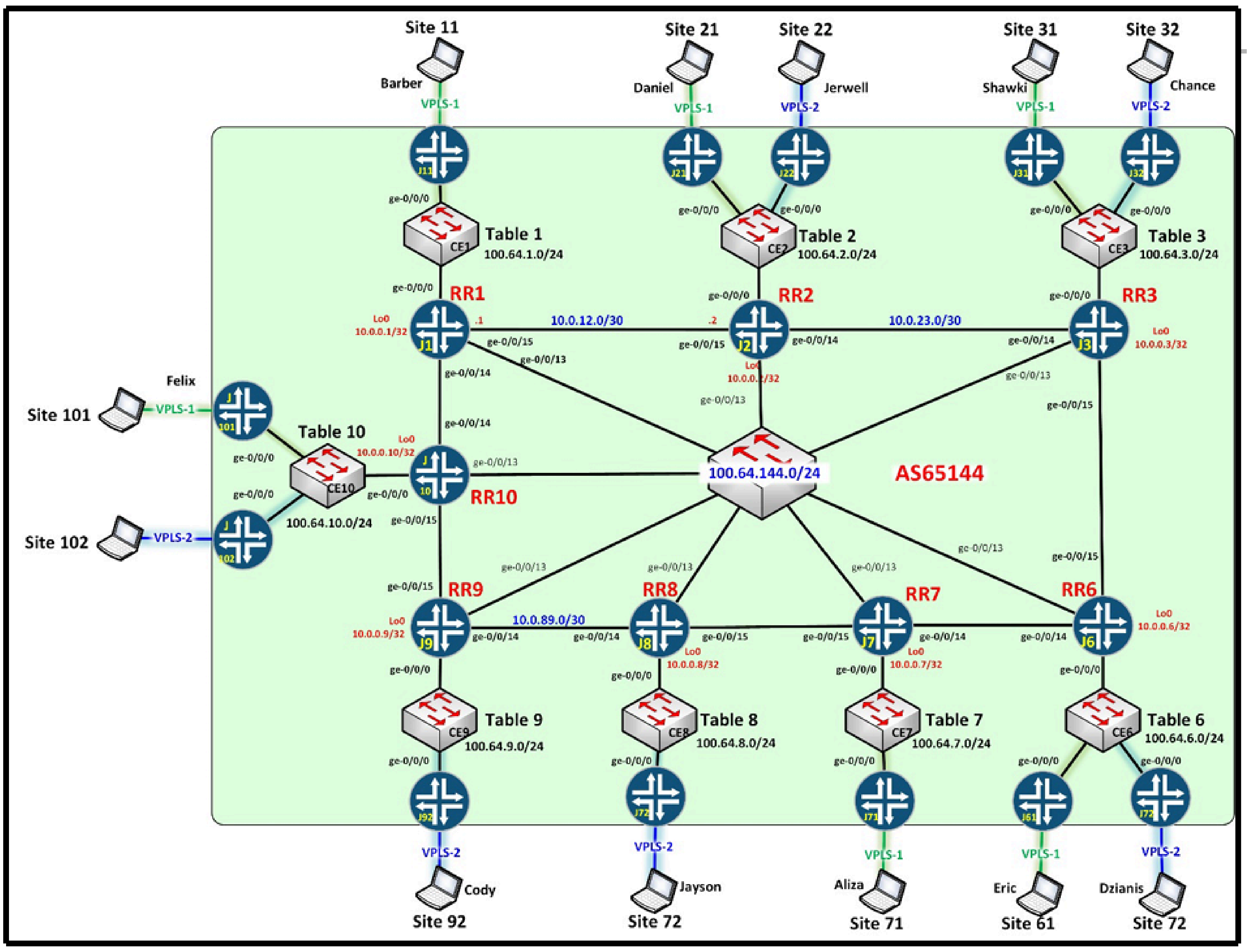
Sheridan College
Faculty of Applied Science & Technology
Professor. Felix Carapaica
Internet Core Technologies
TELE34660

Table of Contents

Introduction

In this assignment, we created a full mesh VPLS L2VPN connection between all the routers for VPLS ID 1 and VPLS ID 2. My personal router was assigned to VPLS ID 2, so I did not configure any settings for VPLS ID 1. During the configuration and testing phase, not everyone completed their setup at the same time as I did. Consequently, only Cody and I were technically connected.

Network Topology



The topology comprised 20 bare-metal Juniper routers within the classroom. Each student was assigned a personal router, while some students shared a table router. Each table router functioned as a route reflector, establishing peer connections with every other route reflector to form a full mesh network.

Preamble

We initially intended to analyze the labels for all the sites, but due to the complexity of the task, we will focus on the labels between Cody and me. VPLS is similar to EVPN VXLAN as it provides a Layer 2 VPN. However, EVPN VXLAN offers several advantages over VPLS. In VPLS, the mechanism is flood-and-learn, whereas in EVPN VXLAN, we benefit from a more advanced control plane that sends updates only when necessary.

Moreover, EVPN VXLAN enhances network efficiency and scalability by reducing broadcast traffic and improving the overall utilization of network resources. It supports larger and more complex network topologies, integrates better with modern data center architectures, and offers enhanced support for multi-tenancy and network segmentation. Additionally, EVPN VXLAN improves network resiliency and reliability with faster convergence times and better fault tolerance.

Another key advantage of EVPN VXLAN is its use of BGP exclusively for signaling, while VPLS can use either LDP (Martini) or BGP (Kompella), making EVPN VXLAN simpler and more streamlined in its signaling approach.

Challenges

One primary challenge was finding the correct labels, requiring us to connect to the switch and scan the interface with Wireshark. Additionally, only Cody and I successfully connected to VPLS ID 2, complicating verification.
Another issue was configuring the site-range command. Initially, we set the VPLS range to 10, assuming it would support 10 sites. However, it only supports site identifiers 1 to 10. Our site identifiers were 32 and 92, so they were out of range. We resolved this by increasing the site-range to 100, allowing site identifiers 1 to 100. This adjustment was necessary, as VPLS relies on the configured range to determine valid site identifiers for establishing pseudowires.

https://www.juniper.net/documentation/us/en/software/junos/cli-reference/topics/ref/statement/site-range-edit-protocols-vpls.html

Personal Router

```
set version 12.3X48-D51
set system host-name J31
set system root-authentication encrypted-password "$1$VA/RFioN$77KTM/lEz9ltLBlBy9bk20"
set security forwarding-options family inet6 mode packet-based
set security forwarding-options family mpls mode packet-based
set interfaces ge-0/0/0 unit 0 family inet address 100.64.3.2/24
set interfaces ge-0/0/0 unit 0 family mpls
set interfaces ge-0/0/15 description "SOME CLIENT"
set interfaces ge-0/0/15 encapsulation ethernet-vpls
set interfaces ge-0/0/15 unit 0 family vpls
set interfaces lo0 unit 0 family inet address 10.0.0.32/32
set interfaces lo0 unit 0 family mpls
set routing-options router-id 10.0.0.32
set routing-options autonomous-system 65144
set protocols mpls interface all
set protocols bgp group S144 type internal
set protocols bgp group S144 local-address 10.0.0.32
set protocols bgp group S144 family l2vpn signaling
set protocols bgp group S144 peer-as 65144
set protocols bgp group S144 neighbor 10.0.0.3
set protocols ospf area 0.0.0.0 interface lo0.0
set protocols ospf area 0.0.0.0 interface ge-0/0/0.0
set protocols ldp interface ge-0/0/0.0
set protocols ldp interface ge-0/0/1.0
set protocols ldp interface all
set protocols ldp interface lo0.0
set routing-instances VPLS-32 instance-type vpls
set routing-instances VPLS-32 interface ge-0/0/15.0
set routing-instances VPLS-32 route-distinguisher 10.0.0.32:2
set routing-instances VPLS-32 vrf-target target:65144:2
set routing-instances VPLS-32 protocols vpls site-range 100
set routing-instances VPLS-32 protocols vpls no-tunnel-services
set routing-instances VPLS-32 protocols vpls site J3 site-identifier 32
set routing-instances VPLS-32 protocols vpls vpls-id 2
```

Personal Router

The main configurations required for my personal router included MPLS, LDP, BGP, and VPLS. For the sake of this explanation, let's assume that the MPLS, LDP, and BGP configurations were correctly set up.
When configuring VPLS, several key steps were involved:

1. Interface Selection: We had to choose which interface would belong to the VPLS instance. This step is crucial as it designates the physical or logical interface that will participate in the VPLS.
2. Route Distinguisher: We needed to set our route distinguisher, which functions similarly to a label. My route distinguisher was 10.0.0.32:2. This helps distinguish between different routes in a multi-tenant environment.
3. Target Configuration: The target specifies the BGP community we want to be a part of. For our setup, the target value was the same for everyone to ensure all participants were in the same BGP community. In our case, the target was 65144:2.
4. VPLS ID: The VPLS ID identifies which VPLS instance we want to be a part of. My VPLS ID was 2, which grouped all devices within the same VPLS.
5. Site Identifier: Each participant needed a unique site identifier. My site identifier was 32.

In summary, configuring VPLS involved selecting the appropriate interface, setting up the route distinguisher, configuring a common target for BGP community membership, assigning a VPLS ID for grouping, and designating a unique site identifier. This setup ensured that all participants could communicate within their designated VPLS group.

Table Router

```
set security forwarding-options family mpls mode packet-based
set interfaces ge-0/0/0 description S144
set interfaces ge-0/0/0 unit 0 family inet address 100.64.144.3/24
set interfaces ge-0/0/0 unit 0 family mpls
set interfaces ge-0/0/1 unit 0 family inet address 100.64.3.1/24
set interfaces ge-0/0/1 unit 0 family mpls
set interfaces ge-0/0/2 unit 0
set interfaces ge-0/0/14 description JERWELL
set interfaces ge-0/0/14 unit 0 family inet address 10.0.23.2/30
set interfaces ge-0/0/14 unit 0 family mpls
set interfaces ge-0/0/15 description ERIC
set interfaces ge-0/0/15 unit 0 family inet address 10.0.56.1/30
set interfaces ge-0/0/15 unit 0 family mpls
set interfaces lo0 unit 0 family inet address 10.0.0.3/32
set routing-options router-id 10.0.0.3
set routing-options autonomous-system 65144
set protocols mpls interface all
set protocols bgp peer-as 65144
set protocols bgp group S144 type internal
set protocols bgp group S144 local-address 10.0.0.3
set protocols bgp group S144 family l2vpn signaling
set protocols bgp group S144 cluster 10.0.0.3
set protocols bgp group S144 neighbor 10.0.0.1
set protocols bgp group S144 neighbor 10.0.0.2
set protocols bgp group S144 neighbor 10.0.0.3
set protocols bgp group S144 neighbor 10.0.0.4
set protocols bgp group S144 neighbor 10.0.0.5
set protocols bgp group S144 neighbor 10.0.0.6
set protocols bgp group S144 neighbor 10.0.0.7
set protocols bgp group S144 neighbor 10.0.0.8
set protocols bgp group S144 neighbor 10.0.0.9
set protocols bgp group S144 neighbor 10.0.0.10
set protocols bgp group S144 neighbor 10.0.0.32
set protocols bgp group S144 neighbor 10.0.0.31
set protocols ospf area 0.0.0.0 interface lo0.0
set protocols ospf area 0.0.0.0 interface ge-0/0/0.0
set protocols ospf area 0.0.0.0 interface ge-0/0/15.0
set protocols ospf area 0.0.0.0 interface ge-0/0/14.0
set protocols ospf area 0.0.0.0 interface ge-0/0/1.0
set protocols ldp interface ge-0/0/0.0
set protocols ldp interface ge-0/0/1.0
set protocols ldp interface ge-0/0/14.0
set protocols ldp interface ge-0/0/15.0
set protocols ldp interface all
```

The table router configurations required LDP, MPLS, and BGP—simple as that! Additionally, we wanted to configure our table router as a route reflector, so we added the cluster 10.0.0.3 to indicate it was a BGP route reflector (RR). Although the router didn't have VPLS, I am assuming we still needed the l2vpn signaling command for BGP to facilitate Layer 2 VPN.

Show VPLS Connections

After configuring VPLS on my personal router, I noticed that I could see both Jayson and Cody's VPLS sites. However, they were marked as "OR" (Out of Range). This indicated that the current site range setting did not permit the VPLS connection to function properly. Consequently, we adjusted the site range to 100, and voilà, it worked.

```
root@J31> show vpls connections
Layer-2 VPN connections:

Legend for connection status (St)
EI -- encapsulation invalid      NC -- interface encapsulation not CCC/TCC/VPLS
EM -- encapsulation mismatch     WE -- interface and instance encaps not same
VC-Dn -- Virtual circuit down    NP -- interface hardware not present
CM -- control-word mismatch      -> -- only outbound connection is up
CN -- circuit not provisioned    <- -- only inbound connection is up
OR -- out of range               Up -- operational
OL -- no outgoing label          Dn -- down
LD -- local site signaled down   CF -- call admission control failure
RD -- remote site signaled down  SC -- local and remote site ID collision
LN -- local site not designated  LM -- local site ID not minimum designated
RN -- remote site not designated RM -- remote site ID not minimum designated
XX -- unknown connection status  IL -- no incoming label
MM -- MTU mismatch               MI -- Mesh-Group ID not available
BK -- Backup connection          ST -- Standby connection
PF -- Profile parse failure      PB -- Profile busy
RS -- remote site standby        SN -- Static Neighbor
LB -- Local site not best-site   RB -- Remote site not best-site
VM -- VLAN ID mismatch

Legend for interface status
Up -- operational
Dn -- down

Instance: VPLS-32
  BGP-VPLS State
  Local site: J3 (32)
    connection-site           Type  St     Time last up          # Up trans
    82  Jayson                rmt   OR
    92  Cody                  rmt   OR
  LDP-VPLS State
  VPLS-id: 2
```

Once the site-range was changed to 100 we were officially connected.

```
Instance: VPLS-32
  BGP-VPLS State
  Local site: J3 (32)
    connection-site           Type  St     Time last up          # Up trans
    82                        rmt   OL
    92                        rmt   Up     Nov 17 22:03:42 2024           1
      Remote PE: 10.9.9.9, Negotiated control-word: No
      Incoming label: 262188, Outgoing label: 262184
      Local interface: lsi.1048576, Status: Up, Encapsulation: VPLS
        Description: Intf - vpls VPLS-32 local site 32 remote site 92
  LDP-VPLS State
  VPLS-id: 2
```

LDP Database

```
root@J31> show ldp database
Input label database, 10.0.0.32:0--10.0.0.3:0
  Label     Prefix
 299840      10.0.0.1/32
 299856      10.0.0.2/32
      3      10.0.0.3/32
 299872      10.0.0.6/32
 299792      10.0.0.7/32
 299904      10.0.0.8/32
 299808      10.0.0.9/32
 299776      10.0.0.10/32
 299936      10.0.0.11/32
 300000      10.0.0.31/32
 299952      10.0.0.32/32
 300048      10.0.0.101/32
 300032      10.5.13.13/32
 300064      10.7.7.7/32
 299920      10.8.8.8/32
 299984      10.9.9.9/32

Output label database, 10.0.0.32:0--10.0.0.3:0
  Label     Prefix
 299840      10.0.0.1/32
 299856      10.0.0.2/32
 299776      10.0.0.3/32
 299872      10.0.0.6/32
```

Using the show ldp database command, we can see all the LDP labels. We can see the input label if someone wants to access 10.0.0.3/32 (label 3, POP), and label 299840 if they want to reach 10.0.0.1/32.
It seems like some people in the class have chosen the wrong address for their router, as 7.7.7, 8.8.8, and 9.9.9 shouldn't exist in our topology. Also, what the heck is 10.5.13.13? Barbarians!

Route Instance Detail

```
root@J31> show route instance detail
master:
  Router ID: 10.0.0.32
  Type: forwarding        State: Active
  Tables:
    inet.0                 : 40 routes (40 active, 0 holddown, 0 hidden)
    inet.3                 : 15 routes (15 active, 0 holddown, 0 hidden)
    mpls.0                 : 23 routes (23 active, 0 holddown, 0 hidden)
    bgp.l2vpn.0            : 9 routes (5 active, 0 holddown, 0 hidden)

VPLS-32:
  Router ID: 0.0.0.0
  Type: vpls              State: Active
  Interfaces:
    lsi.1048576
    ge-0/0/15.0
  Route-distinguisher: 10.0.0.32:2
  Vrf-import: [ __vrf-import-VPLS-32-internal__ ]
  Vrf-export: [ __vrf-export-VPLS-32-internal__ ]
  Vrf-import-target: [ target:65144:2 ]
  Vrf-export-target: [ target:65144:2 ]
  Fast-reroute-priority: low
  Tables:
    VPLS-32.l2vpn.0        : 12 routes (8 active, 0 holddown, 0 hidden)

__juniper_private1__:
  Router ID: 0.0.0.0
  Type: forwarding        State: Active
  Interfaces:
    sp-0/0/0.16383
    lo0.16385
  Tables:
    __juniper_private1__.inet.0: 9 routes (7 active, 0 holddown, 0 hidden)

__juniper_private2__:
  Router ID: 0.0.0.0
  Type: forwarding        State: Active
  Interfaces:
    lo0.16384
  Tables:
    __juniper_private2__.inet.0: 1 routes (0 active, 0 holddown, 1 hidden)

__master.anon__:
  Router ID: 0.0.0.0
  Type: forwarding        State: Active
```

Inside the route instance detail, we can see that our VPLS-32 instance has 12 routes, 8 of which are active. We can also see our route imports and exports and what VPLS interface we are using.

Route Forwarding Table

```
root@J31> show route forwarding-table family vpls
Routing table: VPLS-32.vpls
VPLS:
Destination        Type RtRef Next hop           Type Index NhRef Netif
default            perm     0                    rjct   550     1
ge-0/0/15.0        user     0                    comp   581     2
lsi.1048832        user     0                    comp   587     5
lsi.1048835        user     0                    comp   587     5
lsi.1048836        user     0                    comp   587     5
lsi.1048837        user     0                    comp   587     5
00:45:1d:1b:a3:b5/48 dynm     0                  ucst   578    11 ge-0/0/15.0
00:50:79:66:68:00/48 dynm     0                  indr 262142     5
                              100.64.3.1        Push 262184, Push 299984(top)   583     2 ge-0/0/0.0
00:87:31:ca:b7:37/48 dynm     0                  ucst   578    11 ge-0/0/15.0
00:b0:e1:36:08:37/48 dynm     0                  ucst   578    11 ge-0/0/15.0
00:b0:e1:36:0d:37/48 dynm     0                  ucst   578    11 ge-0/0/15.0
00:b6:70:2f:56:b5/48 dynm     0                  ucst   578    11 ge-0/0/15.0
68:2c:7b:f9:f9:36/48 dynm     0                  ucst   578    11 ge-0/0/15.0
68:3b:78:8a:a7:bf/48 dynm     0                  ucst   578    11 ge-0/0/15.0
84:8a:8d:e0:ae:b5/48 dynm     0                  ucst   578    11 ge-0/0/15.0
88:a4:c2:5e:f1:b2/48 dynm     0                  indr 262145     4
                              100.64.3.1        Push 262160, Push 300144(top)   596     2 ge-0/0/0.0
b4:96:91:31:e7:53/48 dynm     0                  indr 262144     4
                              100.64.3.1        Push 262216, Push 300032(top)   592     2 ge-0/0/0.0
b4:96:91:31:e9:d8/48 dynm     0                  indr 262143     4
                              100.64.3.1        Push 262192, Push 300096(top)   588     2 ge-0/0/0.0
b4:96:91:31:ea:a8/48 dynm     0                  indr 262142     5
                              100.64.3.1        Push 262184, Push 299984(top)   583     2 ge-0/0/0.0
d4:2c:44:ed:ce:0e/48 dynm     0                  ucst   578    11 ge-0/0/15.0
```

When we run show route forwarding-table family vpls, we can see all the VPLS instances. In our case, we have only one instance (VPLS-32). This command shows all the MAC addresses that we are sharing and receiving. MAC addresses with a next hop of 0 are those present in my router. The ones with a next hop of 100.64.3.1 are MAC addresses within the VPLS connection, indicating that the next hop is the table router.

We can see I have more addresses because I was sharing the Sheridan Network MAC devices from my VPLS interface, while Cody was only using GNS3 to share some devices with me. The MAC address **00:50:79:66:68:00 is his GNS3 Virtual PC**.

Incoming Label Calculation

The label calculations for VPLS can be confusing, but I will run through a simulation of the labels between my site 32 and Cody's site 92. At first, I thought I needed access to Cody's router to see his base labels, but in reality, I do not. There is a simple command you can run to figure out all the labels for all sites.

show vpls connections extensive local-site 32

```
Instance: VPLS-32
  BGP-VPLS State
  Local site: J3 (32)
    Number of local interfaces: 1
    Number of local interfaces up: 1
    IRB interface present: no
    ge-0/0/15.0
    lsi.1049094         62        Intf - vpls VPLS-32 local site 32 remote site 62
    lsi.1049095         82        Intf - vpls VPLS-32 local site 32 remote site 82
    lsi.1049088         92        Intf - vpls VPLS-32 local site 32 remote site 92
    Label-base        Offset     Size  Range     Preference
    262265            17         8      6         100
    Label-base        Offset     Size  Range     Preference
    262169            33         8      8         100
    Label-base        Offset     Size  Range     Preference
    262225            57         8      6         100
    Label-base        Offset     Size  Range     Preference
    262233            81         8      2         100
    Label-base        Offset     Size  Range     Preference
    262241            89         8      4         100
    Label-base        Offset     Size  Range     Preference
    262249            97         8      6         100
```

Based on these tables, I should figure out what my incoming label is from Cody's **site 92**. First, we can calculate the label using the formula: **Label = Label Base + (ID - Offset)**.
Therefore, his **ID is 92**. Now, what are his **offset and base?** When we look at the local site labels, we can see an offset of **89** and an offset of **97**. **Cody's ID 92 is smaller than 97**, so it can't be that one; it must be the offset of **89**. Since it’s the offset of **89**, the label base must be **262241**.
Let's now do the calculation: **Label = 262241 + 92 - 89 Label = 262244**
Therefore, my incoming label from Cody is 262244.
Lets check
show vpls connections extensive remote-site 92
We can see from that command our **outgoing** label is **262232** and **incoming** label is **262244**, a match!

```
Instance: VPLS-32
    Remote site: 92
    Number of local interfaces: 0
    Number of local interfaces up: 0
    IRB interface present: no
    Label-base        Offset     Size  Range     Preference
    262169            9          0      8         100
    Label-base        Offset     Size  Range     Preference
    262241            17         0      8         100
    Label-base        Offset     Size  Range     Preference
    262225            25         0      8         100
    Label-base        Offset     Size  Range     Preference
    262201            57         0      8         100
    Label-base        Offset     Size  Range     Preference
    262217            81         0      8         100
    Label-base        Offset     Size  Range     Preference
    262161            89         0      8         100
    Label-base        Offset     Size  Range     Preference
    262209            97         0      8         100
    connection-site           Type  St     Time last up          # Up trans
    J3 (32)                   rmt   Up     Nov 22 19:30:03 2024           1
      Remote PE: 10.9.9.9, Negotiated control-word: No
      Incoming label: 262244, Outgoing label: 262232
      Local interface: lsi.1049088, Status: Up, Encapsulation: VPLS
        Description: Intf - vpls VPLS-32 local site 32 remote site 92
    Connection History:
        Nov 22 19:30:03 2024  status update timer
        Nov 22 19:30:03 2024  loc intf up                  lsi.1049088
        Nov 22 19:30:03 2024  PE route changed
        Nov 22 19:30:03 2024  Out lbl Update                    262232
        Nov 22 19:30:03 2024  In lbl Update                     262244
        Nov 22 19:30:03 2024  loc intf down
```

Outgoing Label Calculation

Since we have done the calculation of our incoming label from Cody now lets do the calculation of the outgoing label towards Cody. We already know what the label should be from the earlier command. The **outgoing** label should be **262232.**

show vpls connections extensive remote-site 92

```
Instance: VPLS-32
    Remote site: 92
    Number of local interfaces: 0
    Number of local interfaces up: 0
    IRB interface present: no
    Label-base        Offset     Size  Range     Preference
    262169            9          0      8         100
    Label-base        Offset     Size  Range     Preference
    262241            17         0      8         100
    Label-base        Offset     Size  Range     Preference
    262225            25         0      8         100
    Label-base        Offset     Size  Range     Preference
    262201            57         0      8         100
    Label-base        Offset     Size  Range     Preference
    262217            81         0      8         100
    Label-base        Offset     Size  Range     Preference
    262161            89         0      8         100
    Label-base        Offset     Size  Range     Preference
    262209            97         0      8         100
```

"Based on these tables, I should figure out what my outgoing label is to Cody's site **92**. First, we can calculate the label using the formula: **Label = Label Base + (ID - Offset)**.
Therefore, my ID is **32**. Now, what are my offset and base? When we look at the remote site labels, we can see an offset of **25** and an offset of **57**. My ID **32** is smaller than **57**, so it can't be that one; it must be the offset of **25**. Since it’s the offset of **25**, the label base must be **262225**.
Let's now do the calculation: **Label = 262225 + 32 - 25 Label = 262232**
Therefore, my outgoing label to Cody is **262232**.

Conclusion

In this assignment, we successfully established a full mesh VPLS L2VPN connection among the routers assigned to VPLS ID 1 and VPLS ID 2. My personal router, assigned to VPLS ID 2, required detailed configurations of MPLS, LDP, BGP, and VPLS, which were carefully executed to ensure proper connectivity.

Although challenges arose, such as finding the correct labels and adjusting the site-range command, these obstacles were overcome with persistence and troubleshooting. The most significant challenge was ensuring only valid site identifiers were used and validating connections between peers, particularly between Cody and me.

The hands-on experience with VPLS configurations provided invaluable insights into network setup and troubleshooting, highlighting the importance of precise configuration and thorough testing. We also learned to navigate issues related to label assignments and site ranges, ensuring effective communication within our VPLS instance.

Overall, this assignment deepened our understanding of VPLS technologies and also enhanced our practical skills in configuring and managing complex network topologies. The successful connection and communication within the VPLS instance showed us the importance of teamwork and meticulous attention to detail in networking projects, you never know if the other side has it configured correctly.